

# Konzeption und Aufbau eines Prüfstandes für taktile Sensoren

**Bachelorarbeit**

Christian Leupolz



**BACHELORARBEIT**

**KONZEPTION UND AUFBAU EINES  
PRÜFSTANDES FÜR TAKTILE SENSOREN**

Freigabe:

Der Bearbeiter:

Unterschriften

Christian Leupolz



Betreuer:

Michael Strohmayer



Der Institutsdirektor

Prof. Dr. G. Hirzinger



Dieser Bericht enthält 94 Seiten, 80 Abbildungen und 2 Tabellen



**Hochschule  
Augsburg** University of  
Applied Sciences

Fakultät für  
Elektrotechnik

## Bachelorarbeit

Mechatronik

Christian Leupolz

## Konzeption und Aufbau eines Prüfstandes für taktile Sensoren

Verfasser der Bachelorarbeit  
Christian Leupolz  
Nebelhornstraße 18  
86854 Amberg  
Telefon +49 1578 441 6605  
Christian.leupolz@hs-augsburg.de

Erstprüfer: Prof. Dr.-Ing. Franz Raps  
Thema erhalten am: 13.01.2012

Hochschule für  
angewandte Wissenschaften –  
Fachhochschule Augsburg  
University of Applied Sciences

An der Fachhochschule 1  
D-86161 Augsburg

Telefon +49 821 55 86-0  
Fax +49 821 55 86-3222  
www.hs-augsburg.de  
info@hs-augsburg.de

Eingereicht am 13.06.2012  
von Christian Leupolz

Studiengang Mechatronik  
Hochschule Augsburg - Fakultät für Elektrotechnik

Prüfer: Prof. Dr.-Ing. Franz Raps

Betreuer: Prof. Dr.-Ing. Franz Raps  
Dipl.-Ing.(FH) Michael Strohmayer M.Sc.(TUM)

Institut für Robotik und Mechatronik  
Deutsches Zentrum für Luft- und Raumfahrt e.V.

# Zusammenfassung

Im Rahmen der vorliegenden Bachelorarbeit wurde ein Prüfstand für taktile Sensoren konzipiert und aufgebaut. Der Prüfstand ermöglicht die Verifizierung und den Vergleich neuartiger taktiler Sensorprototypen. Dies ermöglicht die Erforschung neuer Sensorwerkstoffe und Sensorgeometrien, sowie eine Kalibrierung erster Sensorprototypen durch die Aufnahme einer Kraft-/Widerstandskurve.

Nach der Recherche ähnlicher Prüfverfahren für die Analyse der menschlichen Haut, wurden die dort verwendeten mechanischen Lösungsansätze auf ihre Eignung als Prüfstand für taktile Sensoren hin evaluiert. Unter Verwendung einer am Institut verfügbaren Zug-/Druckprüfmaschine des Herstellers Zwick/Roell konnten erste Kraftmessungen kompletter taktiler Sensorelemente durchgeführt werden. Die gewonnenen Messdaten dienten als Grundlage für die Auslegung eines Mehrachslinearsystems und dessen Antrieb.

Unter Berücksichtigung der Vor- und Nachteile der möglichen Linearachskombinationen wurde ein Portalsystem mit drei Freiheitsgraden als Prüfstand entworfen. Hierfür wurden Lineareinheiten des Herstellers Mitsuami angefordert und passende Aktoren ausgewählt.

Aufgrund von Lieferverzögerungen musste im zeitlichen Rahmen der Bachelorarbeit auf eine kleinere Lineareinheit zurückgegriffen werden. Mit dieser war die Realisierung der verschiedenen Prüfarten mechanisch, wie auch elektrisch und softwaretechnisch möglich. Nach der Fertigstellung einer Benutzeroberfläche unter Zuhilfenahme der grafischen Programmier-Software LabVIEW® konnten erste Prüfversuche durchgeführt werden. Die implementierten Prüfvarianten funktionierten während den gefahrenen Messungen erwartungsgemäß und stellten die Funktion des Prüfstandes unter Beweis.

# Danksagung

An dieser Stelle bedanke ich mich bei Herrn Prof. Dr.-Ing. Franz Raps für die Betreuung von Seiten der Hochschule. Von Seiten des Instituts für Robotik und Mechatronik des Deutschen Zentrums für Luft- und Raumfahrt bedanke ich mich besonders bei Herrn Dipl.-Ing. (FH) Michael Strohmayer M.Sc.(TUM) für die engagierte Unterstützung, Betreuung und fachliche Inspiration. Für zahlreiche technische Hilfestellungen bedanke ich mich zudem bei den Mitarbeitern der Mechanikwerkstatt und der Elektronikwerkstatt. Des Weiteren möchte ich mich bei Denis Schneider, Florian Schnös, Sven Grob und Daniel Pourroy, für die produktive und angenehme Zusammenarbeit bedanken. Insbesondere danke ich meiner Mutter Reinhilde Leupolz und meinem Vater Johann Leupolz, die mich stets während meines Studiums unterstützt haben.

# Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>I</b>
<b>Danksagung</b>	<b>II</b>
<b>Tabellenverzeichnis</b>	<b>V</b>
<b>Abbildungsverzeichnis</b>	<b>VI</b>
<b>Quellcodeverzeichnis</b>	<b>VIII</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Aufgabenstellung . . . . .	1
1.2 Rahmenbedingungen . . . . .	2
1.3 Problemstellung . . . . .	2
<b>2 Stand der Technik</b>	<b>3</b>
2.1 Forschungsstand taktiler Sensorsysteme . . . . .	3
2.2 Prüfmethoden zur Charakterisierung der menschlichen Haut . . . . .	3
2.2.1 R. Lundström: Lokale Vibrationen - Mechanischer Widerstand der unbehaarten menschlichen Haut . . . . .	3
2.2.2 H. Egekvist: Kraftmessung eines Nadeleinstiches beim Menschen . . . . .	4
2.2.3 M. F. Bruce und D. C. Sinclair: Zusammenhang zwischen der taktilen Reizschwelle und der Histologie des menschlichen Fingers . . . . .	4
2.3 Mehrachssysteme . . . . .	5
2.3.1 Lineartechnik . . . . .	5
2.3.1.1 Rundwelle . . . . .	5
2.3.1.2 Unterstützte Rundwelle . . . . .	6
2.3.1.3 Schwalbenschwanzführung . . . . .	6
2.3.1.4 Linearschiene . . . . .	7
2.3.1.5 Kombination zu einem Mehrachsliniersystem . . . . .	7
2.3.2 Linearantriebe . . . . .	9
2.3.2.1 Zahnriemenantrieb . . . . .	9
2.3.2.2 Zahnstangenantrieb . . . . .	9
2.3.2.3 Trapezgewindeantrieb . . . . .	10
2.3.2.4 Kugelumlaufantrieb . . . . .	10
2.3.3 Robotertechnik . . . . .	10
2.4 Aktoren . . . . .	10
2.4.1 Schrittmotor . . . . .	11
<b>3 Entwicklung und Realisierung der Mechanik und Elektronik</b>	<b>13</b>
3.1 Mechanik . . . . .	13
3.1.1 Vorrausgehende Druckversuche taktiler Sensorelemente . . . . .	13
3.1.2 Planung des Mehrachsliniersystems . . . . .	14
3.1.3 Auswahl der Mechanikkomponenten . . . . .	15

3.1.3.1	Auslegung Lineareinheiten . . . . .	15
3.1.3.2	Auslegung Energieführungsketten . . . . .	19
3.1.3.3	Auslegung Kupplungen . . . . .	21
3.1.4	Antriebsauswahl und Auslegung . . . . .	23
3.1.5	Testaufbau für erste Versuche vereinfachter Prüfvorschriften . . . . .	24
3.2	Elektronik . . . . .	24
3.2.1	Steuerungselektronik Mehrachslinearsystem . . . . .	24
3.2.2	Auswahl Messelektronik . . . . .	26
3.2.3	Abschirmung der Messumgebung vor elektromagnetischen Störfeldern . . . . .	27
3.2.4	Evaluierung möglicher Sensorschnittstellen . . . . .	31
<b>4</b>	<b>Software</b>	<b>32</b>
4.1	Steuerungssoftware Mehrachslinearsystem . . . . .	32
4.2	Auswahl einer geeigneten Messsoftware . . . . .	33
4.3	Prüfvorschriften . . . . .	34
4.3.1	Prüfarten . . . . .	34
4.3.2	Implementierung der Prüfvorschriften . . . . .	34
4.4	Benutzeroberfläche des Prüfstandes . . . . .	38
4.5	Verarbeitung und Aufbereitung der Messdatenprotokolle . . . . .	44
<b>5</b>	<b>Messergebnisse</b>	<b>45</b>
5.1	Aufnahme von Messdaten . . . . .	45
5.2	Auswertung der Messergebnisse . . . . .	46
5.2.1	Offsetkompensation Feinwaage: Ermittlung der Kennlinie . . . . .	46
5.2.2	Offsetkompensation Feinwaage: Überprüfung auf Funktion . . . . .	47
5.2.3	Druckversuch von Sensorbahnen verschiedener Geometrie . . . . .	48
<b>6</b>	<b>Ergebnis</b>	<b>50</b>
<b>7</b>	<b>Diskussion</b>	<b>52</b>
<b>8</b>	<b>Ausblick</b>	<b>54</b>
	<b>Literaturverzeichnis</b>	<b>55</b>
	<b>Anhang</b>	<b>57</b>
A	Mechanik Formelsammlung . . . . .	57
B	EMF Schirmungsblech technische Zeichnungen . . . . .	61
C	Quellcode Steuerungssoftware "Stepper Control" . . . . .	64
D	Quellcode Matlabskript zur Messwertaufbereitung . . . . .	71
E	LabVIEW® Blockschartplan Funktionsgruppen . . . . .	72
F	LabVIEW® SubVIs Blockschartpläne und Symbole . . . . .	78
G	Kuka Leichtbauroboter-Struktur Abmessungen . . . . .	81
H	Internationale Vergleichstabelle für Baustähle . . . . .	81
	<b>Eidesstattliche Erklärung</b>	<b>82</b>



# Tabellenverzeichnis

3.1	Vorhandene Messgeräte . . . . .	27
3.2	Neuanschaffung Messgeräte . . . . .	27

# Abbildungsverzeichnis

2.1	Mechanischer Aufbau zur Hautstimulation nach [14, S. 139]	3
2.2	Messapparatur und Graph eines Nadeleinstiches nach [8, S. 43]	4
2.3	Prüftechnik der menschlichen taktilen Reizschwelle nach [5, S. 236]	4
2.4	Rundwelle mit Sinterbronzegleitlager	5
2.5	Linearkugellager	5
2.6	Unterstützte Rundwelle	6
2.7	Linearkugellager für unterstützte Rundwellen	6
2.8	Schemazeichnung Schwalbenschwanzführung	7
2.9	Linearschiene mit Linearwagen [17, S. 567]	7
2.10	Linearschiene und Wagen Funktionsskizze [17, S. 567]	7
2.11	Schematischer Kreutztisch	8
2.12	Typische Kombinationsmöglichkeiten von drei Linearachsen	8
2.13	Schrittmotor Wicklungsansteuerung Prinzipzeichnung [3]	11
3.1	Druckversuch taktiler Sensor DLR mit 8x8 Taxeln [25]	13
3.2	Kuka LBR Struktur	14
3.3	Fünfachsmechanik Beispiel 1	14
3.4	Fünfachsmechanik Beispiel 2	15
3.5	Linearschiene KUT2005L Skizze Querschnittsfläche Abmessungen	16
3.6	Linearschiene Kreutztisch Biegebeanspruchung	17
3.7	Linearschiene X-Achse Montagevariante 2 Biegebeanspruchung	17
3.8	Linearschiene Z-Achse Biegebeanspruchung	18
3.9	Energiekette Querschnitt Leitungen	20
3.10	Energiekettenauslegung Anschlag links	21
3.11	Energiekettenauslegung Anschlag rechts	21
3.12	Testaufbau für Versuche erster Prüfabläufe	24
3.13	Schrittmotorsteuerkarte SMC-TR-1000 [10]	25
3.14	AVR STK500 Starter Kit and Development System [4, S. 2-2]	26
3.15	EMF-Schirmungsblech Hauptelement mit Rückwand	28
3.16	EMF BNC Schaltplan	30
4.1	Ablaufdiagramm "Stepper Control" Hauptschleife	33
4.2	Prüfarten taktile Sensoren und Materialeigenschaften	34
4.3	LabVIEW® Blockschaltplan Leitungsfarben/Datentypen	35
4.4	LabVIEW® Blockschaltplan Strukturen	35
4.5	LabVIEW® Blockschaltplan Flankenerkennung	35
4.6	LabVIEW® Blockschaltplan Übersicht Hauptschleife	36
4.7	LabVIEW® Startbildschirm Benutzeroberfläche Prüfstand	38
4.8	LabVIEW® Benutzeroberfläche Einstellungen	39
4.9	LabVIEW® Benutzeroberfläche Bedienelemente Steuerung	40
4.10	LabVIEW® Benutzeroberfläche grafische Anzeige und Auswertung	42
4.11	Ablaufdiagramm Messdatenverarbeitung mit MATLAB®	44
5.1	Digitalmultimeter Fluke 45 Widerstandsmessung Genauigkeit [11]	45
5.2	MATLAB® Plot Zugversuch Offsetkurve Feinwaage	46
5.3	MATLAB® Plot Zugversuch Vergleich Wegkompensation Feinwaage	47

5.4	MATLAB® Plot Zugversuch Wegkompensation Feinwaage Detailansicht . . . . .	47
5.5	Skizze Druckversuch von Sensorbahnen verschiedener Geometrie . . . . .	48
5.6	MATLAB® Plot Druckversuch taktile Sensorbahn rund . . . . .	49
5.7	MATLAB® Plot Druckversuch taktile Sensorbahn dreieckig . . . . .	49
6.1	Testaufbau Übersicht . . . . .	50
6.2	Prüfhalterungen und ihre Anwendung . . . . .	51
A.1	Biegefälle Tabelle Teil 1 [15, A14.1] . . . . .	57
A.2	Biegefälle Tabelle Teil 2 [15, A14.1] . . . . .	58
A.3	Flächenmomente 2. Grades [15, A7.1] . . . . .	59
A.4	Elastizitätsmodul E [15, A14.1] . . . . .	60
B.1	EMF Schirmungsbelch Hauptrahmen . . . . .	61
B.2	EMF Schirmungsblech Rückseite . . . . .	62
B.3	EMF Schirmungsblech Deckel . . . . .	63
E.1	LabVIEW® Blockschaltplan „Ansteuerung Schrittmotor Messung“ . . . . .	72
E.2	LabVIEW® Blockschaltplan „Ansteuerung Schrittmotor fahre zu Wert“ . . . . .	72
E.3	LabVIEW® Blockschaltplan „Ansteuerung Schrittmotor fahre zu Wert“ . . . . .	72
E.4	LabVIEW® Blockschaltplan „Waage Wert auslesen“ . . . . .	73
E.5	LabVIEW® Blockschaltplan „Digitalmultimeter Wert auslesen“ . . . . .	73
E.6	LabVIEW® Blockschaltplan „Thermometer Wert auslesen“ . . . . .	73
E.7	LabVIEW® Blockschaltplan „Motorschritte in Weg umrechnen“ . . . . .	74
E.8	LabVIEW® Blockschaltplan „Offsetkorrektur / Dehnung “ . . . . .	74
E.9	LabVIEW® Blockschaltplan „Messwertanzeige Graphen, Messwertprotokoll erstellen“ . . . . .	75
E.10	LabVIEW® Blockschaltplan „Rücksetzen der seriellen Schnittstellen“ . . . . .	76
E.11	LabVIEW® Blockschaltplan „Vorzeichen für Zug-/Druckversuch“ . . . . .	76
E.12	LabVIEW® Blockschaltplan „Berechnung/Anzeige des Messfortschritts“ . . . . .	76
E.13	LabVIEW® Blockschaltplan Schrittmotor manuelle Steuerung . . . . .	77
E.14	LabVIEW® Blockschaltplan Schleifen beenden . . . . .	77
F.1	LabVIEW® SubVI „Stepper Control“ Symbol . . . . .	78
F.2	LabVIEW® SubVI „Stepper Control“ Blockschaltplan . . . . .	78
F.3	LabVIEW® SubVI „SerialTemperatureP600“ Symbol . . . . .	79
F.4	LabVIEW® SubVI „SerialTemperatureP600“ Blockschaltplan . . . . .	79
F.5	LabVIEW® SubVI „Offsetkorrektur“ Symbol . . . . .	79
F.6	LabVIEW® SubVI „Offsetkorrektur“ Blockschaltplan . . . . .	80
F.7	LabVIEW® SubVI „Dehnung ZV“ Symbol . . . . .	80
F.8	LabVIEW® SubVI „Dehnung ZV“ Blockschaltplan . . . . .	80
G.1	Kuka LBR Struktur Abmessungen . . . . .	81
H.1	Internationale Vergleichstabelle für Baustähle [12] . . . . .	81

## Quellcodeverzeichnis

C.1	Ausführliche Dokumentation Steuerungssoftware "Stepper Control" . . . . .	64
C.2	Steuerungssoftware "Stepper Control" vollständig . . . . .	67
D.1	Matlabskript "Auswertung Zugversuch" . . . . .	71

# 1 Einleitung

Die sich schnell weiterentwickelnde Digitaltechnik treibt im gleichen Maße auch die Robotertechnik immer weiter voran. Vor allem in der Industrie schreitet die zunehmende Automatisierung von Prozessen weiter fort. Immer häufiger können komplizierte Arbeitsschritte, die bisher nur von Menschen durchgeführt werden konnten, durch Mehrachsrobotersysteme ebenfalls erledigt werden.

Nicht nur die Industrie profitiert von den immer besser werdenden Robotersystemen, sondern auch für die Anwendung in Krankenhäusern zur minimal-invasiven Chirurgie werden bereits Robotersysteme wie das da Vinci® System von Intuitive Surgical® am Deutschen DaVinci® Zentrum (DDVZ) erfolgreich eingesetzt. Ein weiteres noch in der Entwicklung befindliches Chirurgiesystem ist das am Deutschen Zentrum für Luft- und Raumfahrt (DLR) entwickelte MiroSurge System.

In Zukunft sollen autonom agierende Roboter auch in Haushalten und Gastronomien als Servicekraft vermehrt eingesetzt werden.

Der Kontakt zwischen Mensch und Maschine wird daher immer stärker, was eine Gefährdung des Menschen durch den Roboter zur Folge haben kann. Um ein sicheres Miteinander zu gewährleisten, müssen Roboter ihre Umwelt möglichst genau wahrnehmen können. Dafür ist eine Vielzahl an Sensorsystemen notwendig. Einer der wichtigsten Sinne für den Menschen zur Wahrnehmung seiner Umwelt ist der Tastsinn. Er ermöglicht es, Oberflächen verschiedenen Materialien zuzuordnen, die Temperatur von Gegenständen zu erfassen und feinfühlig motorische Bewegungen durchzuführen. Vor allem aber können durch schnelle Reaktion auf Kontakt Ereignisse Verletzungen vermieden werden. Wären Robotersysteme ebenfalls flächendeckend mit solchen Sensoren ausgestattet, könnte durch Integration von Schutzverhaltensmustern ein Arbeiten des Roboters zusammen mit den Menschen gefahrlos realisiert werden.

Diese Tastsensoren werden in der Robotik als taktile Sensoren bezeichnet. Derzeit werden am Institut für Robotik und Mechatronik des Deutschen Zentrums für Luft- und Raumfahrt taktile Sensoren für Robotersysteme entwickelt. Um die Entwicklung dieser Sensoren voranzutreiben, sind Messaufbauten zur Bestimmung der Sensoreigenschaften notwendig.

## 1.1 Aufgabenstellung

Für die Entwicklung neuer, polymer-basierter, taktiler Sensoren ist es notwendig, die hergestellten Prototypen auf ihre mechanischen und elektrischen Eigenschaften zu prüfen. Des Weiteren gibt es bisher keinen standardisierten Test, der taktile Sensoren verschiedener Hersteller untereinander vergleichbar macht.

Im Rahmen der Bachelorarbeit sollen verschiedene Prüfkonzpte erarbeitet werden, deren Hauptaugenmerk vorerst auf der statischen Prüfung für die Prototypenentwicklung taktiler Sensoren liegt. Die Bewertung der verschiedenen Prüfkonzpte geschieht durch die physiologischen Anforderungen an taktile Sensoren. Anhand dieser Bewertung ist das geeignetste Prüfverfahren auszuwählen und ein entsprechender Teststand umzusetzen.

### Aufgaben:

- Evaluierung verschiedener Prüfkonzeppte und Auswahl
- Ausarbeitung des ausgewählten Prüfkonzepptes
- Erstellung einer Stückliste und Bestellung
- Inbetriebnahme des Prüfstandes mechanisch
- Entwurf der Prüfsoftware und Prüfvorschrift
- Fertigstellung des Prüfstandes
- Dokumentation des Prüfstandes

## 1.2 Rahmenbedingungen

Die Aufgabenstellung wird im Rahmen einer Bachelorarbeit des Fachbereiches Elektrotechnik der Hochschule Augsburg in Zusammenarbeit mit dem Institut für Robotik und Mechatronik des Deutschen Zentrums für Luft- und Raumfahrt in Oberpfaffenhofen bearbeitet. Grundlage für die Aufgabenstellung ist ein institutseigener taktiler Sensor, anhand dessen erste Versuche zu Materialeigenschaften und Signalanalysen durchgeführt werden sollen. Aufbauend auf diesen Erkenntnissen soll eine Auslegung des Prüfstandes erfolgen. Der zeitliche Rahmen für die Erstellung verschiedener Prüfkonzeppte, deren Auswahl und Bewertung und die Umsetzung in einen Prüfstand beträgt zwei Mannmonate.

## 1.3 Problemstellung

### Anforderungen an den Prüfstand:

- Sensorgrößen ca.  $25cm \times 15cm \times 12cm$  siehe Abb. G.1
- Taxelanzahl 64 (Sensormesspunkte auf Sensorfläche)
- Wegauflösungsvermögen der statischen Anregung im 1/100mm Bereich
- Schutz gegenüber Störquellen (z.B. Elektromagnetische Störfelder)
- Aufzeichnung der Messbedingungen (z.B. Temperaturverlauf)
- Orientierung der Sensorflächen im Raum (Sensor auf Roboterstruktur)

## 2 Stand der Technik

### 2.1 Forschungsstand taktiler Sensorsysteme

Die nachfolgende Auflistung enthält typische Funktionsprinzipien taktiler Sensorsysteme. Tiefergehende Informationen können den Quellen [25], [24], [23], [6], [27], [13] entnommen werden.

- Resistiv
- Kapazitiv
- Induktiv
- Magnetisch
- Piezoelektrisch
- Pneumatisch
- Hydraulisch
- Optisch

Das langfristige Ziel der Entwicklung von taktilen Sensorsystemen für Roboter ist es die mechanischen und sensorischen Eigenschaften der menschlichen Haut nachzuempfinden. Daher wird im Folgenden auf Prüfmethode zur Analyse der Eigenschaften der menschlichen Haut eingegangen.

### 2.2 Prüfmethode zur Charakterisierung der menschlichen Haut

#### 2.2.1 R. Lundström: Lokale Vibrationen - Mechanischer Widerstand der unbehaarten menschlichen Haut

Um eine Vorhersage menschlicher Reaktionen auf Vibrationen treffen zu können, baute Lundström einen Apparat, der es ihm ermöglichte, eine definierte Anregung mit bestimmter Frequenz, Stärke, Dauer, Richtung, Kontaktpunkt und Anpresskraft aufzubringen. Damit gelang es ihm, den mechanischen Widerstand der Haut zu bestimmen.[14, S. 137ff.]

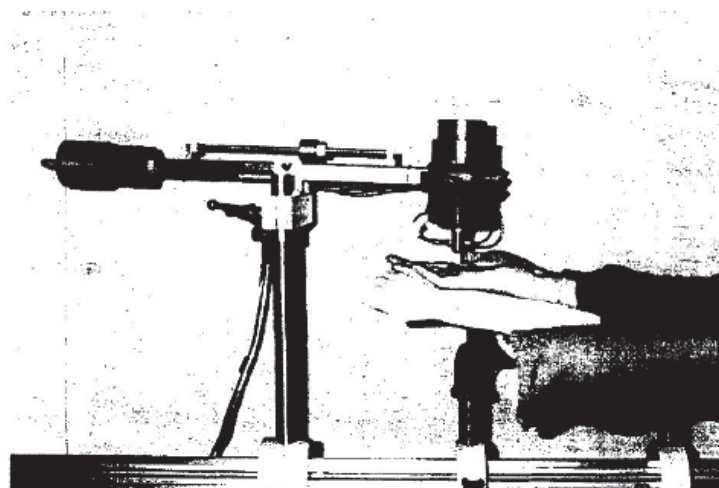


Abbildung 2.1: Mechanischer Aufbau zur Hautstimulation nach [14, S. 139]





## 2.3 Mehrachssysteme

### 2.3.1 Lineartechnik

Die Lineartechnik ist eine der Hauptbereiche der Antriebstechnik. Sie beschreibt Führungssysteme, die sich auf translatorische Bewegungen beschränken. Diese Führungen sind zum Beispiel in Werkzeugmaschinen, im industriellen Anlagenbau, sowie in konventionellen Tintenstrahldruckern vorzufinden.

#### 2.3.1.1 Rundwelle

Eines der einfachsten Linearsysteme ist die Rundwelle mit Lineargleitlager oder Linearkugellager. Es besteht aus einer geschliffenen Präzisionswelle, auf die zum Beispiel ein Gleitlager aus Sinterbronze aufgeschoben wird, siehe Abb. 2.4. Um eine Rotation um die Welle zu vermeiden, sind mindestens zwei parallele Lineareinheiten, deren Schlitten miteinander starr verbunden sind, nötig.



Abbildung 2.4: Rundwelle mit Sinterbronzegleitlager

Eine höherwertige Ausführung gegenüber den Lineargleitlagern sind die Linearkugellager. Sie bestehen aus einer Metallbuchse, die Führungsbahnen für Kugeln beinhaltet. Die Linearbewegung ist selbst bei hohen Traglasten durch die Abrollbewegung der Kugeln leichtgängig. Siehe Abb. 2.5



Abbildung 2.5: Linearkugellager

### 2.3.1.2 Unterstützte Rundwelle

Große Längen oder höhere Traglasten können bei einfachen Rundwellen zu einer starken Durchbiegung der Welle führen, da diese nur an den Wellenenden fest gelagert sind. Es wäre möglich, Wellen mit größerem Durchmesser zu verwenden, jedoch sind diese teurer und benötigen mehr Platz bei der Installation. Eine einfachere Lösung bieten unterstützte Rundwellen. Sie sind längs auf ein Aluminium-V-Profil aufgebracht, welches wiederum über die gesamte Länge verschraubt wird. Siehe Abb. 2.6



Abbildung 2.6: Unterstützte Rundwelle

Die unterstützten Rundwellen benötigen angepasste Lineargleit- oder Kugellager mit einem Öffnungswinkel für das Alustützprofil, siehe Abb. 2.7. Bei einer hängenden Montage sollte beachtet werden, dass sich die Traglast durch die fehlende Gleitfläche oder Kugelreihe verringert.



Abbildung 2.7: Linearkugellager für unterstützte Rundwellen

### 2.3.1.3 Schwalbenschwanzführung

Der Name kommt von dem gegabelten Schwanz der Schwalbe, an den diese Trapezführung erinnert. Diese Führung ist meist als Gleitführung vorzufinden. Sie besteht aus geschliffenen Präzisionsstahlteilen, eines in Form des Schwalbenschwanzes und das andere als Negativ, siehe Abb. 2.8. Die Rotation um die Linearachse ist im Gegensatz zu den Rundwellenführungen aufgrund der Schwalbenschwanzform ausgeschlossen. Diese Führung wird aufgrund ihrer großen Masse und den daraus resultierenden guten Dämpfungseigenschaften oft in Metall zerspanenden Maschinen eingesetzt.

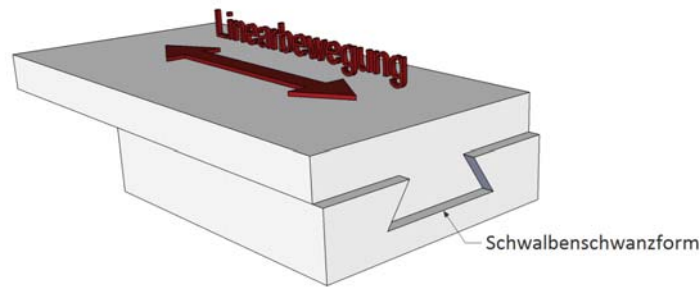


Abbildung 2.8: Schemazeichnung Schwalbenschwanzführung

#### 2.3.1.4 Linearschiene

Kommt es auf hohe Präzision an, werden meist Linearschienen mit Linearwägen gewählt. Wie auch bei den Schwalbenschwanzführungen ist bei einer einzelnen Linearschiene keine Rotation um die eigene Achse möglich, siehe Abb. 2.9.



Abbildung 2.9: Linearschiene mit Linearwagen [17, S. 567]

Wie in nachfolgender Abb. 2.10 zu sehen ist, rollen die Kugeln während einer Linearbewegung zwischen Wagen und Schiene ab, bis sie an einem der beiden Enden ankommen und über den Kunststoffkäfig wieder zurück zum Anfang des Wagens befördert werden.

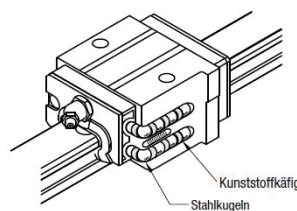


Abbildung 2.10: Linearschiene und Wagen Funktionsskizze [17, S. 567]

#### 2.3.1.5 Kombination zu einem Mehrachslinearsystem

Die einfachste Variante zwei Lineareinheiten miteinander zu verbinden ist die eines Kreuztisches. Mit einem Aufbau wie in Abb. 2.11 zu sehen, können automatisiert beliebige Punkte einer

X-Y-Ebene angefahren werden. In Kombination mit einer dritten Achse in Z-Richtung, wie in Abb. 2.12 (a) zu sehen, ist es möglich die einzelnen Messpunkte taktiler Sensorelemente nacheinander abzufahren und zu prüfen.

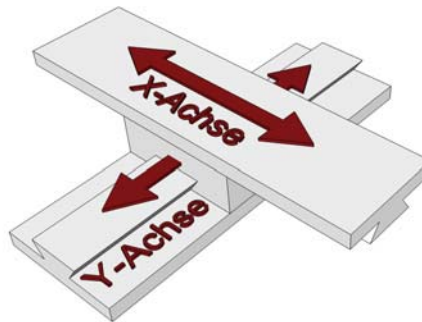
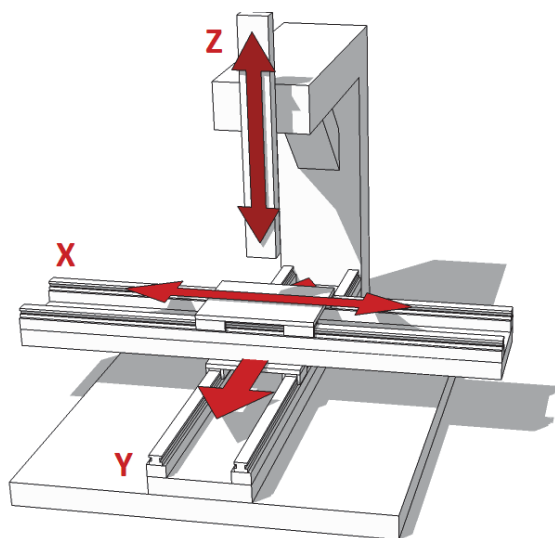
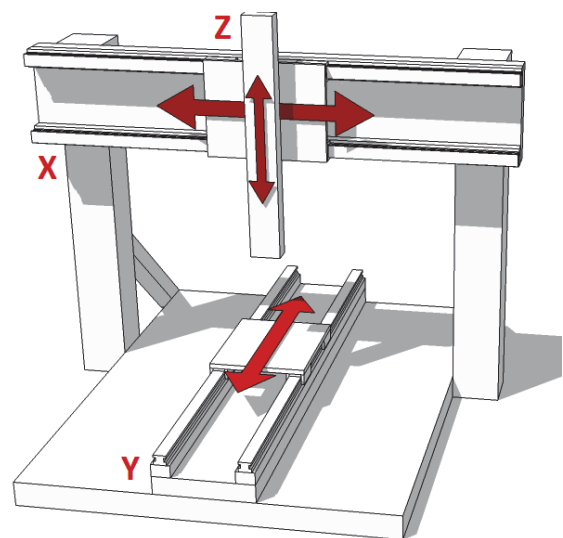


Abbildung 2.11: Schematischer Kreuztisch



(a) Variante 1



(b) Variante 2

Abbildung 2.12: Typische Kombinationsmöglichkeiten von drei Linearachsen

In der nachfolgenden Liste wird auf die Vor- und Nachteile der beiden Linearachskombinationen aus Abb. 2.12 eingegangen.

Vorteile Variante 1:

- X und Y-Achse liegen in parallelen Ebenen
- Weniger Ausrichtungsaufwand
- Weniger Konstruktionsaufwand

Nachteile Variante 1:

- Mehr Durchbiegung der X-Achse
- Erhöhter Platzbedarf bei großen Prüflingen
- Eingeschränkter Fahrweg in der Y-Achse für große Prüflinge

Vorteile Variante 2:

- Weniger Platzbedarf bei großen Prüflingen
- Keine Einschränkung der Fahrwege
- Weniger Durchbiegung der X-Achse

Nachteile Variante 2:

- Mehr Konstruktionsaufwand
- Mehr Ausrichtungsaufwand

### 2.3.2 Linearantriebe

Die Auswahl des Linearantriebs hängt stark vom Einsatzzweck ab. Die Hauptaufgabe eines Linearantriebes ist es, die Rotationsbewegung eines Aktors in eine Translationsbewegung umzuwandeln. Nachfolgend findet sich eine Auflistung häufig verwendeter Linearantriebe, siehe auch [16], [17] und [18].

#### 2.3.2.1 Zahnriemenantrieb

Vorteile:

- Kostengünstig
- Hohe Verfahrgeschwindigkeiten

Nachteile:

- Auflösung hängt stark von Drehwinkelauflösung und Spiel des Antriebes ab
- Geringere Antriebskraft
- Gefahr durch Schwingungen des Zahnriemens

Merkmale:

- Antrieb sowohl feststehend mit geschlossenem umlaufenden Zahnriemen, wie auch mitfahrend bei offenem feststehenden Zahnriemen
- Großzahl an Anwendungsfällen durch verschiedene Zahnprofile wie z.B.: T, AT, HTD, Zoll
- Erreichbare Wegauflösung im 1/10mm - 1/100mm Bereich

Beispiele: Tintenstrahldrucker, Scanner

#### 2.3.2.2 Zahnstangenantrieb

Vorteile:

- Kostengünstig
- große Kräfte
- beliebig lange Fahrwege

Nachteile:

- Auflösung hängt stark von Drehwinkelauflösung und Spiel des Antriebes ab
- Zusätzliches Spiel durch wechselnden Zahnradengriff bei Richtungswechsel

Merkmale:

- Antrieb mitfahrend auf Lineareinheit für große Zahnstangenlängen und oft feststehend für kurze Zahnstangenlängen
- Erreichbare Wegauflösung im 1/10mm Bereich

Beispiel: Zahnradbahn

### 2.3.2.3 Trapezgewindeantrieb

Vorteile:

- Kostengünstig
- Verschiedene Übersetzungen durch unterschiedliche Steigungen
- Hohe Vorschubkräfte erreichbar

Nachteile:

- Erhöhter Verschleiß durch Reibung an den Muttern
- Spiel zwischen Trapezgewinde und Mutter
- Eingeschränkte Spindellängen

Merkmale:

- Antrieb feststehend an Trapezgewindespindelende
- Muttern meist aus Sinter-Bronze oder Nylatron
- Erreichbare Wegauflösung 1/10mm - 1/100mm Bereich (bei gegeneinander verspannten Muttern)

Beispiel: Wagenheber mit Kurbel

### 2.3.2.4 Kugelumlaufantrieb

Vorteile:

- Spielfrei einstellbar
- große Vorschubkräfte
- Sehr hohe Wegauflösung

Nachteile:

- Kostenintensiv
- Eingeschränkte Spindellängen

Merkmale:

- Erreichbare Wegauflösungen für geschliffene Kugelumlaufspindeln im 1/1000mm Bereich
- Antrieb sowohl feststehend an Kugelumlaufspindelende, wie auch mitfahrend auf Linearschlitzen als Direktantrieb der Kugelumlaufmutter

Beispiel: Linearantrieb Werkzeugmaschinen

### 2.3.3 Robotertechnik

Für Anwendungsfälle, die sechs oder mehr Achsen benötigen werden typischerweise hochpräzise Robotersysteme eingesetzt (z.B. DLR LBR, als Leichtbauroboter allerdings weniger präzise als größere Kuka oder Stäubli Industrieroboter). Die Kosten moderner Robotersysteme überschreiten jedoch das Budget dieser Arbeit und stehen im Auswahlprozess daher nicht zur Verfügung.

## 2.4 Aktoren

Die Aktorauswahl wird auf elektrische Antriebe angesichts flexiblerer Aufstellungsorte des Prüfstandes eingeschränkt. (Grundlegende Informationen dieses Kapitels wurden bezogen aus [21])

Häufig verwendete Aktoren sind:

- Schrittmotoren
- Servomotoren(AC/DC)
- Linearmotoren
- Piezoelektrische Antriebe

Aufgrund der am Institut verfügbaren Schrittmotoren wird im Folgenden detaillierter auf diesen Antrieb eingegangen.

### 2.4.1 Schrittmotor

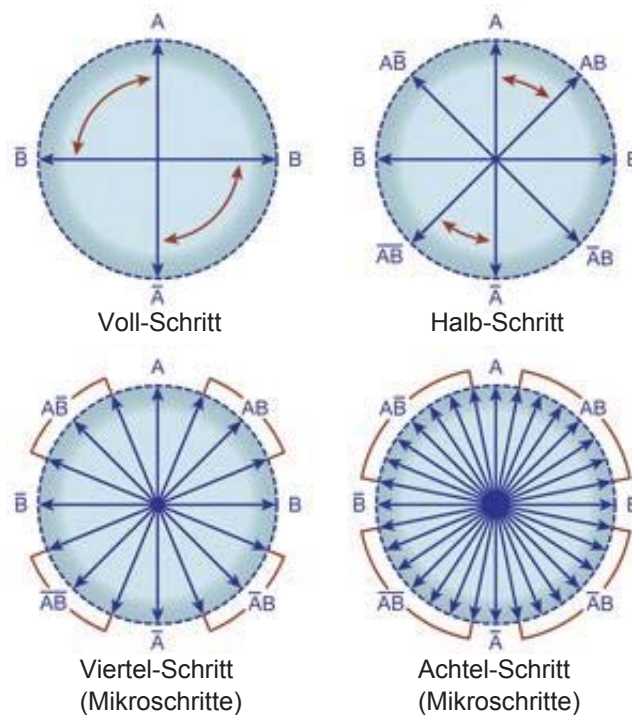


Abbildung 2.13: Schrittmotor Wicklungsansteuerung Prinzipzeichnung [3]

Schrittmotoren sind bürstenlose Gleichstrommotoren, deren Position und Beschleunigung gesteuert werden kann, ohne dafür eine zusätzliche externe Positionsrückmeldung zu benötigen. Sie bestehen typischerweise aus einem rotierenden Permanentmagneten (Rotor), der von Elektromagneten (Stator) umschlossen ist. Wenn die Elektromagneten in richtiger Polarität und Reihenfolge bestromt werden, wird ein Moment erzeugt, das den Rotor in Schritten bewegt. Diese Einzelschritte können aufeinanderfolgend den Motor in verschiedenen Drehzahlen, entsprechend der Wechselrate der Schritte, drehen. Auf die Position kann durch Überwachung der Anzahl der durchgeführten Schrittreihenfolgen rückgeschlossen werden.

Indem man den Strom in den einzelnen Motorphasen genau reguliert, ist es möglich einen ganzen Schrittwinkel eines Schrittmotors in kleinere Mikroschritte zu unterteilen. Der Mikroschrittbetrieb ermöglicht eine höhere Positionsauflösung, vermindert Resonanzprobleme und verringert somit Betriebsgeräusche, siehe [3].

In Abb. 2.13 ist die Ansteuerung der beiden Elektromagneten für die normalen Betriebsmodi „Voll-Schritt“ und „Halb-Schritt“ und für den Mikroschrittbetrieb bis „Achtel-Schritt“ dargestellt.

Vorteile:

- Keine komplexe Regelung erforderlich
- Kostengünstig
- Einfache Ansteuerung (z.B. Schritt / Richtung)
- Wartungsarm (Bürstenlos)

Nachteile:

- Schrittverluste im Fehlerfall (Überlast)
- Mögliche Resonanzfrequenzen

Für die gelisteten Nachteile des Schrittmotorantriebs ist jedoch zu beachten, dass Schrittverluste nur bei einer fehlerhaften Auslegung der dynamischen Antriebsdrehmomente, der Beschleunigungswerte oder bei zu niedriger Versorgungsspannung auftreten. Besonders zu beachten ist, dass die Versorgungsspannung meist viel höher als die angegebene Nennspannung zu wählen ist, da ansonsten der Nennstrom, wegen den Gegeninduktivitäten der Motorwicklungen bei hohen Drehzahlen, nicht aufrecht erhalten werden kann. Die höhere Versorgungsspannung wird ermöglicht durch die Stromregelung der Schrittmotortreiberstufen.

Das Problem der Resonanzfrequenzen kann meist durch den Mikroschrittbetrieb behoben werden.



## 3 Entwicklung und Realisierung der Mechanik und Elektronik

### 3.1 Mechanik

#### 3.1.1 Vorrausgehende Druckversuche taktiler Sensorelemente

Um erste Aussagen zu den mechanischen Anforderungen an den Prüfstand treffen zu können, müssen die Materialeigenschaften des vorliegenden taktilen Sensors in Druckversuchen auf einer Zwick/Roell Material-Prüfmaschine [30] ermittelt werden. Abb. 3.1 zeigt das Weg/Kraft Diagramm des vorliegenden taktilen Sensorelements, wobei der mechanische Messbereich des Sensors bei ca. 1,5mm endet. An diesem Punkt wird eine Kraft von 57N auf die gesamte Sensorfläche gemessen. Da der Sensor 64 Messpunkte besitzt, wirkt auf einen einzelnen Sensormesspunkt eine Kraft von 0,89N. Dies ist folglich die Maximalprüfkraft, die der geplante Prüfstand mindestens aufbringen können muss.

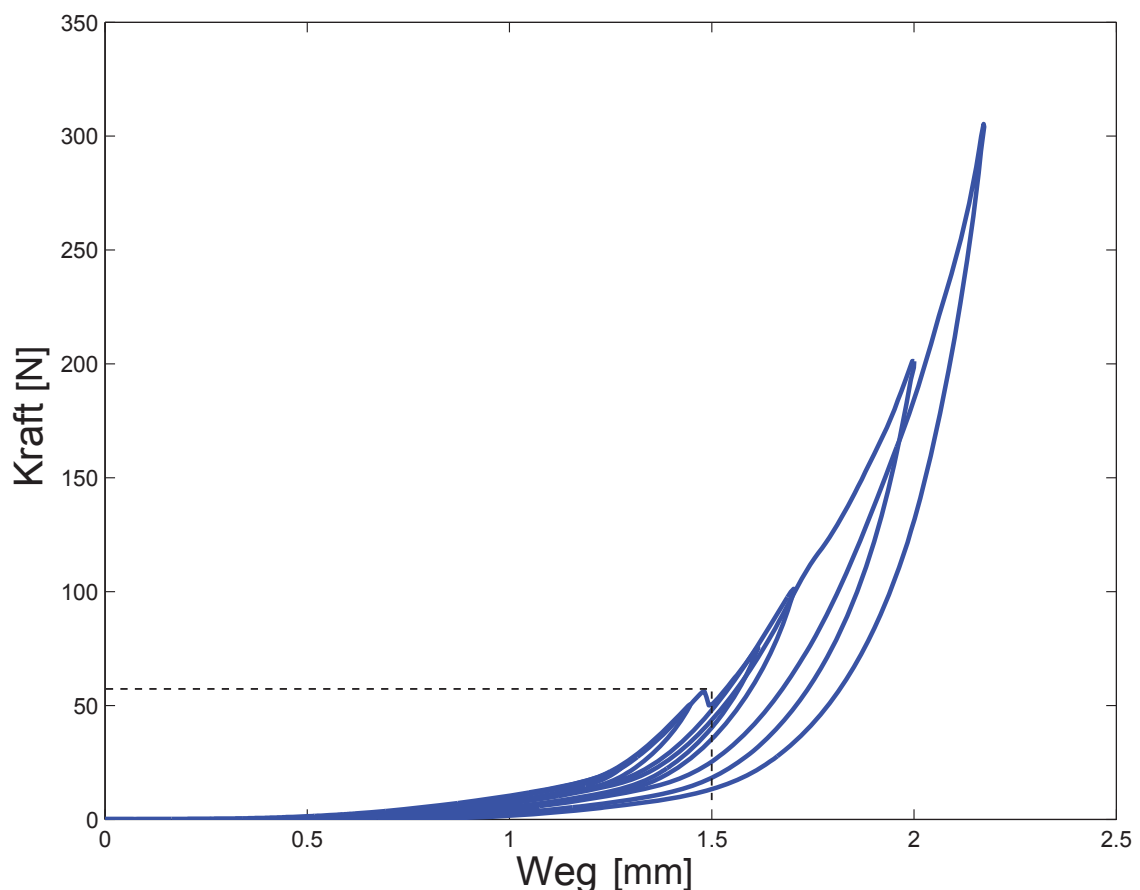


Abbildung 3.1: Druckversuch taktiler Sensor DLR mit 8x8 Taxeln [25]

### 3.1.2 Planung des Mehrachslinearsystems



Eine Anforderung an das Linearsystem ist die Erweiterbarkeit auf bis zu fünf Achsen, da in Zukunft auch Sensorelemente direkt auf ihrer angebrachten dreidimensionalen Struktur, wie z.B. eines Roboterarmchassis oder einer Kuka Leichtbau-roboter-Struktur (LBR), siehe Abb. 3.2, geprüft werden.

Um dies zu ermöglichen sind mindestens fünf Freiheitsgrade erforderlich. In Abb. 3.3 ist schematisch eine mögliche Variante einer Fünfachsmechanik dargestellt, wie sie zum Beispiel häufig in zerspanenden Bearbeitungszentren vorzufinden ist. Für

Abbildung 3.2: Kuka LBR Struktur

geringere Kräfte und kleinen Werkzeugabmessungen werden meist Achskombinationen, wie in Abb. 3.4 zu sehen ist, eingesetzt.

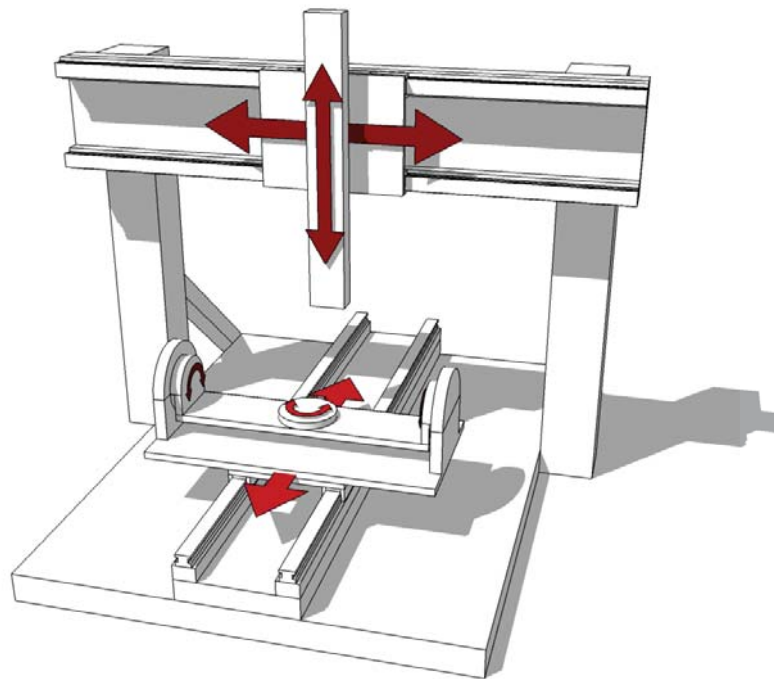


Abbildung 3.3: Fünfachsmechanik Beispiel 1

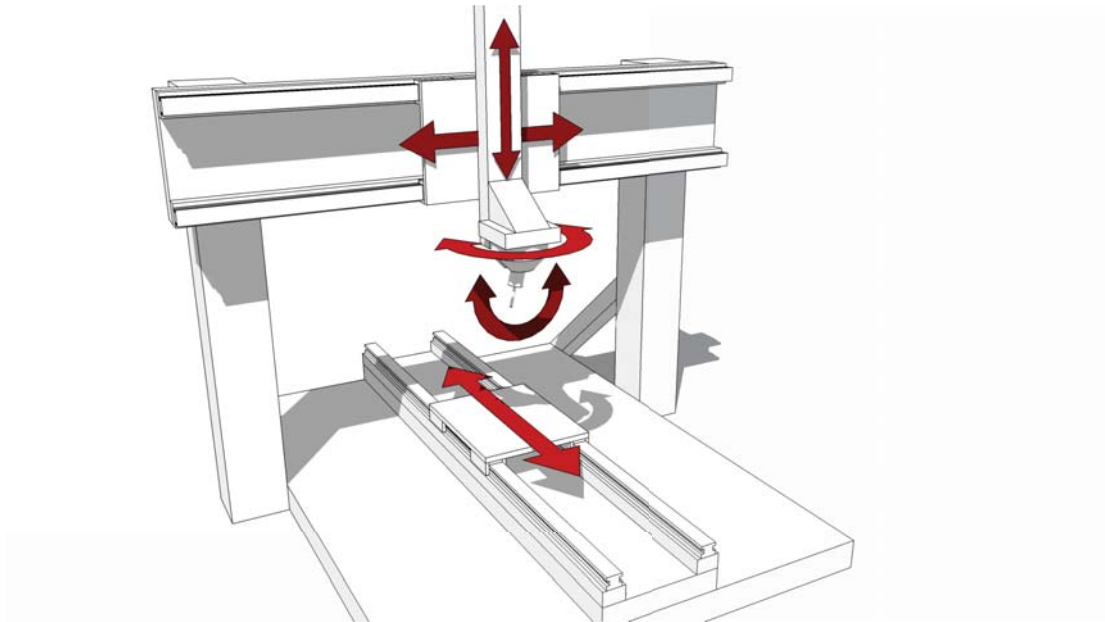


Abbildung 3.4: Fünfachsmechanik Beispiel 2

### 3.1.3 Auswahl der Mechanikkomponenten

#### 3.1.3.1 Auslegung Lineareinheiten

Eine Präzisionslineareinheit besteht meist aus einer Linearschiene und einem Linearwagen. Der Wagen besitzt auf der Schiene nur einen translatorischen Freiheitsgrad, kann daher nur auf einer Linie vor und zurück bewegt werden.

Bei der Auswahl der Linearschienen sind folgende Punkte zu beachten:

- Wiederholgenauigkeit: Maximaler Positionsfehler beim Anfahren der gleichen Position
- Positioniergenauigkeit: Maximaler Positionsfehler über gesamten Verfahrensweg
- Flächenträgheitsmoment: Ausrichtung der Achse, Durchbiegung
- Spindelsteigung: Auflösung, Kraft, Verfahrensgeschwindigkeit
- Anlaufdrehmoment: Entscheidend für Antriebsauswahl
- Zulässige statische und dynamische Traglast: Wie viel Gewicht kann an die Linearachse montiert werden
- Zulässiges statisches Moment: Welche Momente dürfen maximal in den einzelnen Rotationsachsen aufgebracht werden
- Massenträgheitsmoment: Bestimmt die maximale Beschleunigung

Der nachfolgende Abschnitt behandelt die wesentlichen Überlegungen und Berechnungen zur Anordnung und Auslegung der Lineareinheiten.

### Benötigte Verfahrswege der Lineareinheiten:

Die minimal benötigten Verfahrswege, sind anhand eines Elements einer Kuka Leichtbauroboter-Struktur, auf die taktile Sensorelemente aufgebracht werden, mit den Abmessungen  $25cm \cdot 15cm \cdot 12cm$ , siehe Anhang Abb. G.1, vorgegeben. Es ist vorgesehen diese auf ihrer Zielstruktur prüfen zu können. Damit alle Messpunkte der taktile Sensorelemente auf dieser Struktur mit einer zur Oberfläche senkrechten Prüfkraft beaufschlagt werden können, ist eine Erweiterung des Prüfstandes um zwei Rotationsachsen nötig und der zusätzlich benötigte Freiheitsraum für diese vorzusehen. Aus diesen Gründen sind folgende Verfahrswege gewählt worden:

Annahme benötigter Platz für Rotationskopf: ca. 20cm

Verfahrsweg  $x/y \approx LBR_{\max} + Rotationskopf_{\text{links}} + Rotationskopf_{\text{rechts}} = 25cm + 2 \cdot 20cm = 65cm$

Verfahrsweg  $z \approx LBR_{\max} + Rotationskopf_{\text{oben}} = 25cm + 20cm = 45cm$

Gewählte Verfahrswege nach Datenblatt [17, S. 465] und [17, S. 422]:

X/Y-Achse (KUT2005L-820-200): 610mm

Z-Achse (LX3005P-S2-T3056.4-600): 467mm

### Anordnung der Linearachsen hinsichtlich ihrer Biegebeanspruchung:

Da das Flächenträgheitsmoment der Lineareinheit KUT2005L weder im Datenblatt angegeben ist, noch über den telefonischen Support der Firma Auskunft darüber geben werden kann, muss dieses näherungsweise berechnet werden.

Für das Material der Grundplatte AlMg0,7SiT6 wird zur Annäherung der mittlere Elastizitätsmodul (E-Modul) von Aluminiumlegierungen aus Anhang Abb. A.4 verwendet.

$$E_{\text{Aluminiumleg.}} = 70kN/mm^2$$

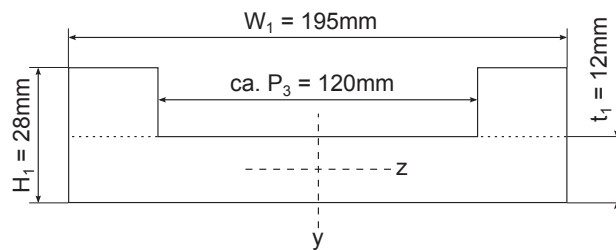


Abbildung 3.5: Linearschiene KUT2005L Skizze Querschnittsfläche Abmessungen

Mit den Angaben aus Abb. 3.5 kann das Flächenträgheitsmoment nach Anhang Abb. A.3 wie folgt berechnet werden.

$$I_z = \frac{1}{3} \cdot (BH^3 + bh^3) - (BH + bh) \cdot e_1^2$$

mit  $B = W_1 - P_3 = 75mm$  /  $H = H_1 = 28mm$  /  $b = P_3 = 120mm$  /  $h = t_1 = 12mm$

$$\text{und } e_1 = \frac{1}{2} \cdot \frac{BH^2 + bh^2}{BH + bh} = \frac{1}{2} \cdot \frac{75mm \cdot (28mm)^2 + 120mm \cdot (12mm)^2}{75mm \cdot 28mm + 120mm \cdot 12mm} = 10,75mm$$

$$\Rightarrow I_z = \frac{1}{3} \cdot (75mm \cdot (28mm)^3 + 120mm \cdot (12mm)^3) - (75mm \cdot 28mm + 120mm \cdot 12mm) \cdot (10,75mm)^2 = 57,99cm^4$$

$$I_y = \frac{1}{12} \cdot (BH^3 - bh^3)$$

mit  $B = H_1 = 28mm$  /  $H = W_1 = 195mm$  /  $b = H_1 - t_1 = 16mm$  /  $h = P_3 = 120mm$

$$\Rightarrow I_z = \frac{1}{12} \cdot (28mm \cdot (195mm)^3 - 16mm \cdot (120mm)^3) = 1499,74cm^4$$

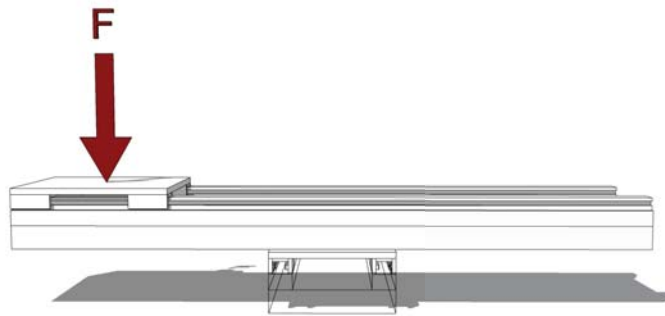


Abbildung 3.6: Linearschiene Kreuztisch Biegebeanspruchung

Annahme: Belastung der X-Achse in Abb. 3.6 mit der Kraft  $F_{\max} = 10kg \cdot 9,81 \frac{m}{s^2} = 98,1N$

$$\Rightarrow \text{Durchbiegung nach Abb. A.1 Nr.1 } w_A = \frac{Fl^3}{3EI} = \frac{98,1N \cdot (610mm/2)^3}{3 \cdot 70kN/mm^2 \cdot 57,99cm^4} = 23\mu m$$

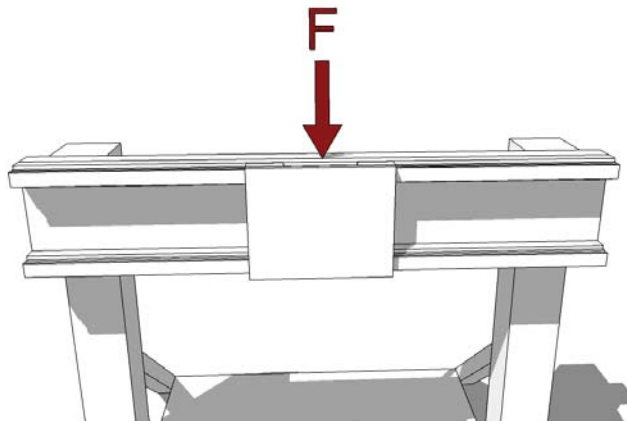


Abbildung 3.7: Linearschiene X-Achse Montagevariante 2 Biegebeanspruchung

Annahme: Belastung der X-Achse in Abb. 3.7 mit der Kraft  $F_{\max} = 10kg \cdot 9,81 \frac{m}{s^2} = 98,1N$

$$\Rightarrow \text{Durchbiegung nach Abb. A.2 Nr.6 } w_m = \frac{Fl^3}{48EI} = \frac{98,1N \cdot (610mm)^3}{48 \cdot 70kN/mm^2 \cdot 1499,74cm^4} = 0,4\mu m$$

⇒ Die Montage der X-Achse nach Abb.3.7 ist aufgrund der wesentlich geringeren Durchbiegung zu wählen.

Angaben Datenblatt Lineareinheit Z-Achse (LX3005P):

Werkstoff Führungskörper [17, S. 421]: JIS STKM

JIS STKM ist nach [12] ein japanischer Baustahl. Ein allgemeiner Kennwert des Elastizitätsmoduls für Baustahl ist  $210\text{kN/mm}^2$ , siehe Anhang Abb. A.4

Flächenträgheitsmoment um X [17, S. 402]:  $I_x = 2,5 \cdot 10^4 \text{ mm}^4$

Annahme: Belastung der Z-Achse im Punkt Abb. 3.8 (a) mit der Kraft  $F_{\max} = 5\text{kg} \cdot 9,81 \frac{\text{m}}{\text{s}^2} = 49,05\text{N}$  im Abstand von 50mm zur Lineareinheit.

⇒ Resultierendes Drehmoment (siehe Abb. 3.8 (b))  $M_Z = 0,05\text{m} \cdot 49,05\text{N} = 2,45\text{Nm}$

⇒ Durchbiegung nach Abb. A.1 Nr.2  $w_A = \frac{Ml^2}{2EI} = \frac{2,45\text{Nm} \cdot (467\text{mm})^2}{2 \cdot 210\text{kN/mm}^2 \cdot 25000\text{mm}^4} = 0,05\text{mm}$

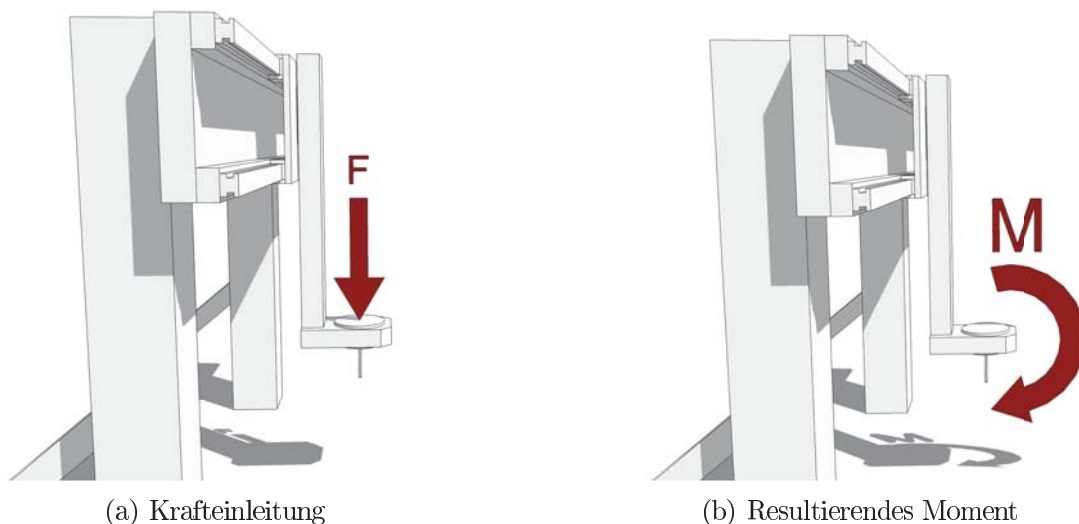


Abbildung 3.8: Linearschiene Z-Achse Biegebeanspruchung

### Vorschubkraft der Lineareinheiten:

Die Vorschubkräfte der X und Y-Achse sind nicht von wesentlicher Bedeutung, da Sie keine Lasten zu heben haben oder Kräfte auf eine Fläche aufbringen müssen, sondern lediglich der Positionierung dienen. Die Vorschubkraft bzw. Hubkraft der Z-Achse ist von Bedeutung, da sie einerseits ihr Eigengewicht und das des montierten Antriebes zu heben hat und andererseits genügend Kraftreserven eingeplant werden müssen, um eine Erweiterung der Z-Achse um zwei zusätzliche Rotationsachsen zu ermöglichen.

Vorschubkraft X/Y-Achse:

Angaben:

Schrittmotor dyn. Drehmoment: ca.  $M_{\text{Haltemoment}}/2 \Rightarrow 2,7Nm/2 = 1,35Nm$

(Annäherung, da dynamisches Drehmoment abhängig von der Drehzahl)

Steigung P Kugelgewindetrieb Lineareinheit KUT2005L: 5mm

$$F_{\text{Vorschub X/Y}} = \frac{M_{\text{Motor dynamisch}}}{P_{\text{Kugelgewindetrieb}}} = \frac{2,7Nm/2}{0,005m} = 270N$$

Hubkraft Z-Achse:

Angaben:

Schrittmotor dyn. Drehmoment: ca.  $M_{\text{Haltemoment}}/2 \Rightarrow 3,1Nm/2 = 1,55Nm$

(Annäherung, da dynamisches Drehmoment abhängig von der Drehzahl)

Steigung P Kugelgewindetrieb Lineareinheit LX3005P: 5mm

$$F_{\text{Vorschub X/Y}} = \frac{M_{\text{Motor dynamisch}}}{P_{\text{Kugelgewindetrieb}}} = \frac{3,1Nm/2}{0,005m} = 310N$$

Alle Angaben zu den Daten können in folgenden Dokumenten nachgeschlagen werden:

Formeln siehe Anhang Abb. A.1 und Abb. A.2 Lineareinheit KUT [17, S. 461ff]

Lineareinheit LX [17, S. 401ff]

Schrittmotoren [26], [22]

### 3.1.3.2 Auslegung Energieführungsketten

Energieketten werden zum mechanischen Schutz für Stromleitungen, Druckluftleitungen, Signalleitungen usw. verwendet. Sie ermöglichen es, Kabel und Schläuche zwischen translatorisch bewegten Mechanikkomponenten zu verlegen. Dabei kann der minimal zulässige Biegeradius der Kabel und Schläuche eingehalten und somit die Lebensdauer verlängert werden.

#### 1. Innerer Querschnitt

Der benötigte Querschnitt der Energiekette wird, wie in Abb. 3.9 dargestellt, anhand der einzelnen Leitungsquerschnitte ermittelt.

⇒ Gewählte Energiekette MHPUS206 [17, S. 642] mit Querschnitt<sub>innen</sub>:  $14mm \cdot 40mm$

(Leitungsquerschnitte siehe [9], [26], [22], [17])

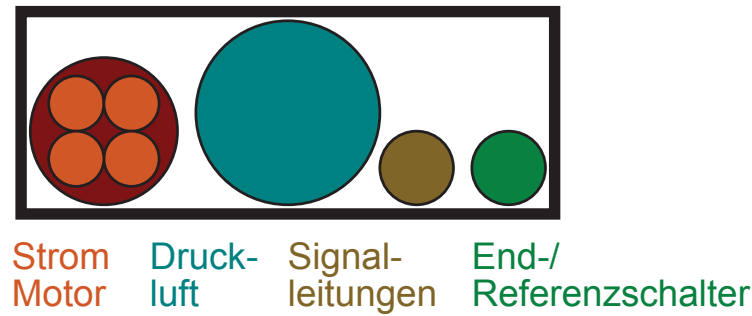


Abbildung 3.9: Energiekette Querschnitt Leitungen

## 2. Biegeradius

Um Leitungsschäden vorzubeugen ist der minimale Biegeradius aller Leitungen zu beachten. Im vorliegenden Fall ist der Biegeradius durch den Druckluftschlauch PU-99003-0604 der Firma Eisele mit 14 mm vorgegeben, siehe [9, S. 7]. Damit der sichere Betrieb bei dynamischer Beanspruchung garantiert werden kann, wird ein Sicherheitsfaktor von 2-3 angenommen. Daraus resultiert ein minimaler Biegeradius von 28-42 mm. Anhand des benötigten Energiekettenquerschnitts und des Biegeradius fiel die Wahl auf die Misumi MHPUS206-38 siehe [17, S.1-642].

## 3. Energiekettenlänge

Die benötigte Länge der Energiekette setzt sich aus zwei Teilen zusammen. Der notwendigen Länge für den halben Umfang des Biegeradius und der benötigten Länge zur Abdeckung des Fahrweges der Lineareinheit, siehe Abb. 3.10 und Abb. 3.11. Anhand der benötigten Gesamtlänge kann über die Gliederabstand die benötigte Kettengliederzahl ermittelt werden.

$$\Rightarrow \Delta L + \frac{1}{2} \cdot (2\pi \cdot r) + 2 \cdot \text{Sicherheit} = 305\text{mm} + 119,4\text{mm} + 200\text{mm} = 624,4$$

mit Gliederabstand [17, S. 642]: 32mm

$$\Rightarrow 624,4\text{mm} \div 32\text{mm} = 19,5 \approx 20 \text{ Kettenglieder}$$

$\Rightarrow$  gewählte Energiekette nach [17, S. 642]: MHPUS206-38-20



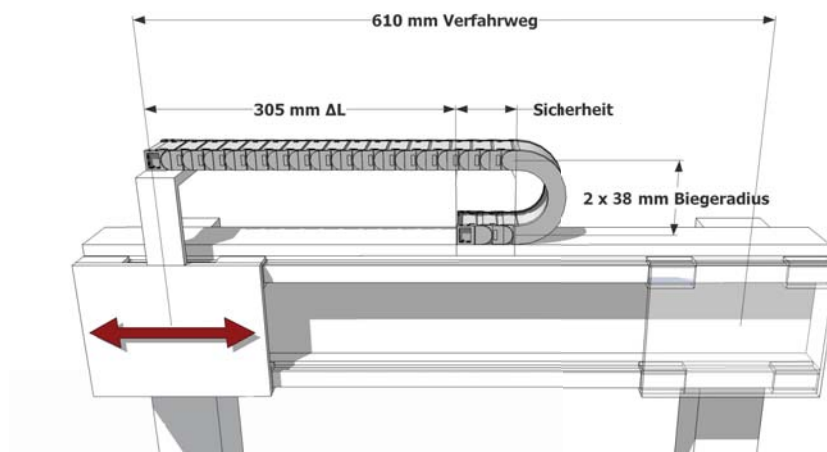


Abbildung 3.10: Energiekettenauslegung Anschlag links

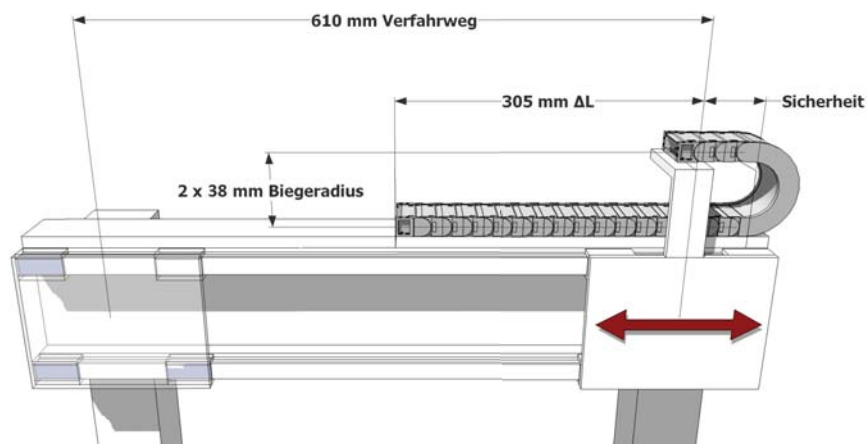


Abbildung 3.11: Energiekettenauslegung Anschlag rechts

### 3.1.3.3 Auslegung Kupplungen

Kupplungen werden im Maschinenbau eingesetzt, um Verbindungen zwischen Antriebswelle und Abtriebswelle herzustellen. Sie dienen dem Ausgleich von Winkelversatz, Radialversatz und Axialspiel. Bei der Auswahl muss vor allem das zulässige Drehmoment berücksichtigt werden. Für Präzisionslineareinheiten mit hoher Positioniergenauigkeit ist die Torsionsfederkonstante ebenfalls zu beachten.

Begriffsklärung:

Torsionsfederkonstante: Gibt an, bei welchem Drehmoment ein entsprechender Winkelversatz zu verzeichnen ist.

Beispiel:

*Torsionsfederkonstante* =  $100\text{Nm/rad}$

*Annahme Drehmoment* =  $1\text{Nm}$

$\Rightarrow 1\text{Nm}/100\text{Nm/rad} = 0,01\text{rad} = 0,01 \cdot (360^\circ/2\pi) = 0,573^\circ$

$((2\pi)\text{rad} = 360^\circ)$

Die folgenden Berechnungen beziehen sich auf die maximalen Haltemomente der verwendeten Schrittmotoren und auf die maximal zulässigen Drehmomente der Kupplungen. Die dynamischen Drehmomente der Schrittmotoren während des Betriebs sind wesentlich geringer und können den Graphen der Datenblätter entnommen werden. Für die Auslegung sind nur die Extrema von Bedeutung.

#### **LINEAREINHEIT KUT2005L:**

- Spindelsteigung 5mm
- Max. Haltemoment Schrittmotor (SANYO DENKI 103H7822-5740) =  $2,7\text{Nm}$

Kupplung SCXW34-8-12

- Maximales Drehmoment  $4,0\text{Nm}$
- Torsionsfederkonstante =  $5700\text{Nm/rad}$

*Maximaler Winkelversatz*

$\Rightarrow 4,0\text{Nm} \div 5700\text{Nm/rad} = 0,000702\text{rad} = 0,040^\circ$

$\Rightarrow 2,7\text{Nm} \div 5700\text{Nm/rad} = 0,000474\text{rad} = 0,027^\circ$

*Resultierender Positionsfehler der Lineareinheit*

$\Rightarrow (0,040^\circ \div 360^\circ) \cdot 5\text{mm} = 0,6\mu\text{m}$

$\Rightarrow (0,027^\circ \div 360^\circ) \cdot 5\text{mm} = 0,4\mu\text{m}$

#### **LINEAREINHEIT LX3005P:**

- Spindelsteigung 5mm
- Max. Haltemoment Schrittmotor (QMOT 6018-86-28-310) =  $3,10\text{Nm}$

Kupplung SCXW21-6-8

- Maximales Drehmoment  $1,2\text{Nm}$
- Torsionsfederkonstante =  $900\text{Nm/rad}$

*Maximaler Winkelversatz*

$\Rightarrow 1,2\text{Nm} \div 900\text{Nm/rad} = 0,00133\text{rad} = 0,076^\circ$

*Resultierender Positionsfehler der Lineareinheit*

$\Rightarrow (0,076^\circ \div 360^\circ) \cdot 5\text{mm} = 1,1\mu\text{m}$

Kupplung MCSLC25-6-8

- Maximales Drehmoment 2,8Nm
- Torsionsfederkonstante = 950Nm/rad

*Maximaler Winkelversatz*

$$\Rightarrow 2,8Nm \div 950Nm/rad = 0,00295rad = 0,169^\circ$$

*Resultierender Positionsfehler der Lineareinheit*

$$\Rightarrow (0,169^\circ \div 360^\circ) \cdot 5mm = 2,3\mu m$$

Alle Angaben zu den Daten können in folgenden Dokumenten nachgeschlagen werden:

Lineareinheit KUT [17, S. 461ff]

Lineareinheit LX [17, S. 401ff]

Kupplungen [17, S. 959ff]

Schrittmotoren [26], [22]

### 3.1.4 Antriebsauswahl und Auslegung

Lineareinheiten mit Spindelantrieb wandeln eine rotatorische in eine translatorische Bewegung um. Es sind daher Aktoren auszuwählen, die eine Rotationsbewegung mit hoher Winkelauflösung bereitstellen, über eine Bremse zum Halten der jeweiligen Position verfügen und über eine vorhandene Energiequelle versorgt werden können.

Da das System möglichst unabhängig von seinem Standort eingesetzt werden soll, wird als Energiequelle Strom vorausgesetzt. Damit verringert sich die Auswahl der Aktoren auf Piezoantriebe, Servomotoren und Schrittmotoren. Nachdem die auftretenden Kräfte des Prüfstandes verhältnismäßig gering sind und Schrittmotoren einfach in der Ansteuerung und preiswert in der Anschaffung sind, fiel die Wahl auf dieses Antriebskonzept.

In der nachstehenden Auflistung sind die gewählten Schrittmotoren zu den jeweiligen Lineareinheiten aufgeführt. Die Schrittmotoren wurden anhand ihrer mechanischen Kennwerte Flanschmaß, Wellendurchmesser, Haltemoment, dynamisches Drehmoment und Masse, sowie ihrer elektrischen Daten Ansteuerung, Phasenstrom, Versorgungsspannung und Wicklungswiderstand ausgewählt.

#### **LINEAREINHEIT LX3005P:**

Schrittmotor Trinamic QMot QSH6018-86-28-310

#### **LINEAREINHEIT KUT2005L:**

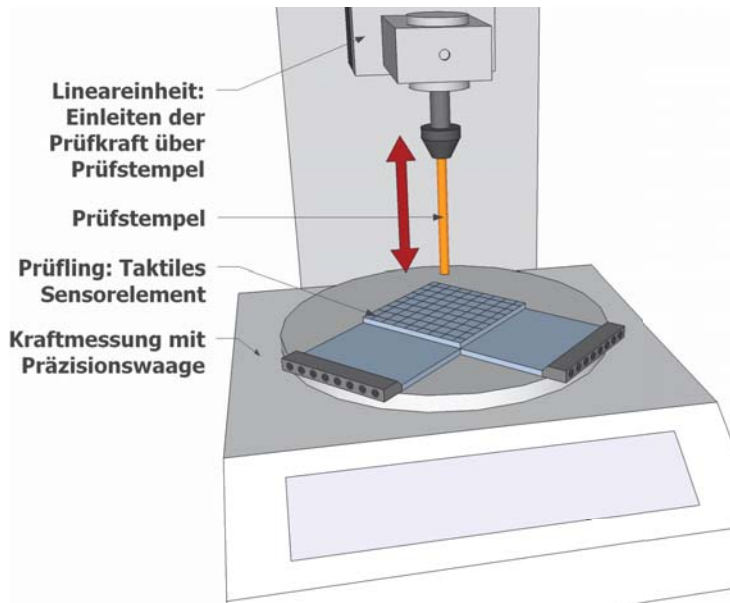
Schrittmotor SANYO DENKI 103H7822-5740

(Lineareinheiten KUT siehe [17, S. 461ff], LX siehe [17, S. 401ff])

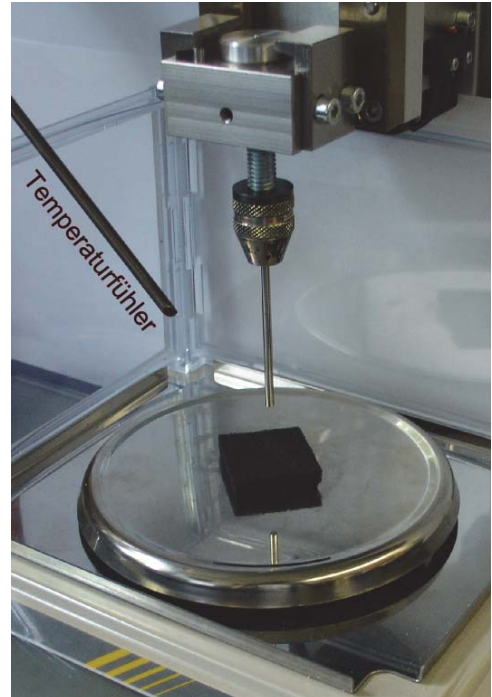
Schrittmotoren siehe [26], [22])

### 3.1.5 Testaufbau für erste Versuche vereinfachter Prüfvorschriften

Aufgrund langer Lieferzeiten der einzelnen Prüfstandkomponenten, werden vereinfachte Prüfverfahren für eine vorhandene, fertige, kleinere Hublineareinheit (MISUMI LX2001 B1 A2025 150 [17, S. 409]) aus einer früheren Abschlussarbeit [19] realisiert, siehe Abb. 3.12. Dadurch kann mit der Implementierung der Prüfverfahren und der Software der Steuerung begonnen werden.



(a) Entwurf



(b) Realisierung

Abbildung 3.12: Testaufbau für Versuche erster Prüfabläufe

## 3.2 Elektronik

### 3.2.1 Steuerungselektronik Mehrachslensystem

Aufgrund nachfolgend gelisteter Kriterien, wurde der IC A4983 von Allegro MicroSystems, Inc. als Treiberbaustein für die Schrittmotoren ausgewählt.

- Komplette Schrittmotortreiberstufe inklusive Logik integriert in einem Baustein
- Mikroschrittfähig bis 1/16 Schritt
- Treiberspannungen bis 35V
- Ströme bis zu 2A
- Stromabsenkung bei Motorstillstand
- Abschaltung bei thermischer Überlast mit Hysterese

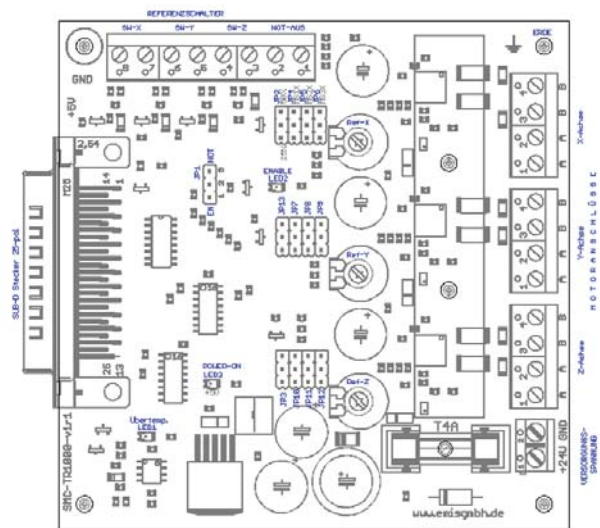
- UVLO (Unterspannungs-Abschaltung)
- Schutz gegenüber Crossover-Strömen
- Kompatible Logikspannungsversorgung und Logikpegel von 3,3V und 5V

(siehe [2, S. 1ff.])

Als fertige Treiberstufe für die drei Schrittmotoren der Hauptlinearachsen X, Y und Z wird auf Grund des verwendeten Treiber IC's Allegro A4983 die 3-Achs-Schrittmotorsteuerkarte SMC-TR-1000 der EMIS GmbH ausgewählt. Ohne Kühlkörper ist ein Phasenstrom bis zu 1A angegeben, auf der Karte ist jedoch die Montage eines Kühlkörpers möglich, wodurch höhere Ströme erzielt werden können. Der Treiberbaustein A4983 von Allegro MicroSystems, Inc. kann laut Datenblatt bei entsprechender Kühlung einen Strom bis zu 2A treiben. Durch die Mikroschrittfähigkeit des A4983 wird eine höhere Winkelauflösung und ein verbesserter Rundlauf erzielt. Die Ansteuerung erfolgt über TTL-Pegel im Schritt-/Richtungsmodus über einen 25 poligen SUB-D Stecker, siehe [10] und [2, S. 2].



(a) Platine



(b) Layout

Abbildung 3.13: Schrittmotorsteuerkarte SMC-TR-1000 [10]

Aufgrund verfügbarer Mikrocontrollerhardware am Institut, wird zur Ansteuerung der Schrittmotorkarte ein AVR STK500 Entwicklungskit mit einem Atmega 16 Mikrocontroller eingesetzt, siehe Abb. 3.14. Das STK500 erfüllt folgende Anforderungen. Die Kommunikation zwischen PC und Mikrocontroller kann über die serielle Schnittstelle erfolgen, da auf dem Entwicklungsboard Pegelwandler und entsprechende Verbindungsstecker zur Verfügung stehen. Die einzelnen Ports des Mikrocontrollers sind alle auf Stiftleisten inklusive Masse und Versorgungspins herausgeführt. Dies eignet sich für die direkte Verbindung mit der Schrittmotorkarte, da hierfür lediglich ein Kabel von einem zehnpoligen Pfostenstecker auf einen 25-poligen SUB-D Stecker angefertigt werden muss. Zusätzlich sind auf dem STK500 je acht Taster und LEDs vorhanden, die für Benutzereingaben und Anzeige verwendet werden können, siehe [4, S. 3-1].

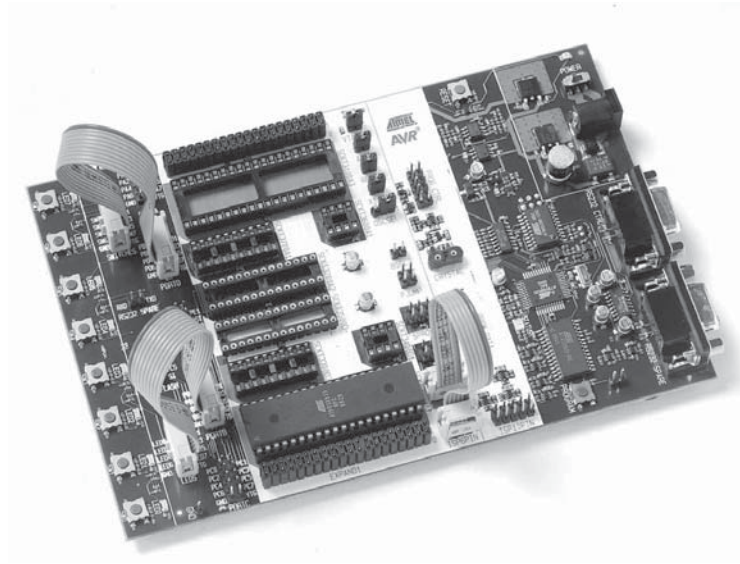


Abbildung 3.14: AVR STK500 Starter Kit and Development System [4, S. 2-2]

### 3.2.2 Auswahl Messelektronik

Hauptkriterien für die Auswahl der Messelektronik sind:

- Hohe Messgenauigkeit
- Hohe Anzahl an Messeingängen (umschaltbar)
- Galvanische Trennung des Messaufbaus von der Messumgebung
- Gleiche Datenschnittstellen (z.B. RS232, GPIB)
- Einfache Ansteuerung über ein gemeinsames Bussystem
- Einfache und schnelle Implementierung in eine Messsoftware (z.B. MATLAB<sup>®</sup>, LabVIEW<sup>®</sup>)
- Möglichkeit einer Signalabschirmung (Messbuchsen der Messgeräte z.B. BNC-Anschluss)
- Verwendung vorhandener Messgeräte (Fluke 45, Frequenzgenerator HP3314A, Oszilloskop Tektronix TDS 510A)

Die Tabelle 3.1 beinhaltet bereits vorhandene Messgeräte, die sich für den geplanten Messaufbau eignen.

Messgerät	Typ	Schnittstellen	Verwendungszweck	Stk.
Fluke 45	Multimeter	RS232/GPIB	Widerstandsmessung	2
Fluke 80TK	Temperaturfühler	für Fluke45	Raumtemperaturmessung	2
Waage Kern EG	Feinwaage	RS232	Kraftaufnahme	1
DOSTMANN P600	präz. Thermometer	RS232	Raumtemperaturmessung	1
NI GPIB-USB-HS	GPIB-USB Adapter	GPIB/USB	Kommunikation GPIB-PC	1
HP3314A	Frequenzgenerator	GPIB	*Anregung Prüfling	1
Tektronix TDS 510A	Oszilloskop 4 ch.	GPIB	*Aufnahme Signalantwort	1

*\*optional*

Tabelle 3.1: Vorhandene Messgeräte

Die Tabelle 3.2 beinhaltet zugekaufte Messgeräte, welche für den geplanten Messaufbau noch benötigt werden.

Messgerät	Typ	Schnittstellen	Verwendungszweck	Stk.
HP 3488A	Modulares Schalt-/ Steuerungs-System	GPIB	Multiplex / I/O	1
Opt.010/HP 44470A	Multiplexer	HP Einschub	Multiplex	2
Opt.012/HP 44472A	VHF Multiplexer BNC	HP Einschub	Multiplex (EMV)	2
Opt.013/HP 44474A	Digitale I/O Karte	HP Einschub	I/O Zugprüfmaschine	2

Tabelle 3.2: Neuanschaffung Messgeräte

### 3.2.3 Abschirmung der Messumgebung vor elektromagnetischen Störfeldern

In der heutigen Zeit sind elektromagnetische Felder (EMF) überall vorhanden. Jedes Mobiltelefon sendet ununterbrochen elektromagnetische Wellen aus, um eine Verbindung mit dem jeweiligen Sendemasten aufrecht zu erhalten. Dutzende Radiostationen senden im Dauerbetrieb rund um die Uhr. Prozessoren in Computern besitzen inzwischen sehr hohe Taktraten, die ebenfalls elektromagnetische Felder erzeugen können. Beinahe jeder Arbeitsplatz und Haushalt besitzt ein drahtloses Netzwerk für die Internetverbindung. Bekannte Störungen, die durch diese Felder verursacht werden können, sind beispielsweise das Knacken und Rauschen in Lautsprecheranlagen während eines eingehenden Mobiltelefonanrufs. Im gleichen Maße können diese Felder auch Messungen verfälschen, weswegen Messumgebungen nach Möglichkeit eine Abschirmung gegenüber diesen Störungen erhalten sollten, siehe [28].

Zu diesem Zweck wird mit dem CAD-Werkzeug Pro/ENGINEER Wildfire 4.0 ein EMF-Schirmungsblech als Blechteilkonstruktion entworfen. In Abb. 3.15 ist die Baugruppe des Hauptbiegeblechrahmens mit eingebauter Rückwand dargestellt. Zur besseren Ansicht wurde der Deckel ausgeblendet. (Informationen zu Blechteil biegen siehe [7, S. 376])

Die zu prüfenden Sensorelemente besitzen zwar lediglich je acht Anschlüsse für die Spalten und Zeilen, jedoch verlaufen die Anschlussbahnen, je nach Sensor, rechtwinklig oder 180° versetzt zueinander. Daher wurden in der Konstruktion des Hauptrahmens drei Seitenwände mit Aussparungen für BNC-Anschlussbuchsen vorgesehen, um die verschiedenen Sensortypen, ohne diese

zu verspannen, kontaktieren zu können.



Abbildung 3.15: EMF-Schirmungsblech Hauptelement mit Rückwand

Wie in Abb. 3.16 zu sehen ist, befindet sich das taktile Sensorelement in einer Art faradayschem Käfig, dessen Potential auf Masse liegt. An drei Seitenwänden des Abschirmungsbleches (siehe Abb. 3.15) befinden sich je acht BNC-Buchsen für die Zeilen und Spalten. Über diese werden die einzelnen Sensoranschlüsse über 16 BNC-Kabel geschirmt zu zwei VHF-Multiplexern HP 44472A geführt. Die Multiplexer befinden sich in Einschüben des modularen Schalt-/Steuerungssystems HP 3488A, welches über die GPIB-Schnittstelle angesteuert wird. Es sind immer vier BNC-Eingangsbuchsen auf eine BNC-Ausgangsbuchse verschaltet, um flexiblere Aufbauten mit den Multiplexern zu ermöglichen. Da hierbei jedoch pro Multiplexereinheit nur ein Signal von Interesse ist, werden die beiden getrennten Common 0 und Common 1 Ausgänge jeweils über BNC-Kabel mit einem T-Stück zusammengeführt. Es liegen nun die über die Multiplexer ausgewählten Sensoranschlüsse auf zwei BNC Anschlüssen vor. Um diese mit dem Digitalmultimeter verbinden zu können, wurde eine spezielle BNC-Adapterbox angefertigt, welche eine Signalleitung auf den inneren Leiter eines BNC Anschlusses führt und die andere Signalleitung mit dem Masseanschluss verbindet. Der BNC Ausgang der Adapterbox ist über ein BNC-Kabel mit einem BNC zu 4mm-Federstecker-Adapter verbunden, welcher an den vorderen Messeingängen des Digitalmultimeters Fluke 45 angeschlossen ist. Diese Maßnahmen sollen die Widerstandsmessungen des Sensorelements vor Störeinkopplungen durch elektromagnetische Felder schützen.



Die Widerstandsmesswerte des Digitalmultimeters werden über dessen GPIB-Schnittstelle abgefragt. Der GPIB Bus wird wiederum über einen GPIB-USB-Adapter, der mit einem Laborrechner verbunden ist, über die grafische Programmierungsumgebung LabVIEW<sup>®</sup> angesteuert.

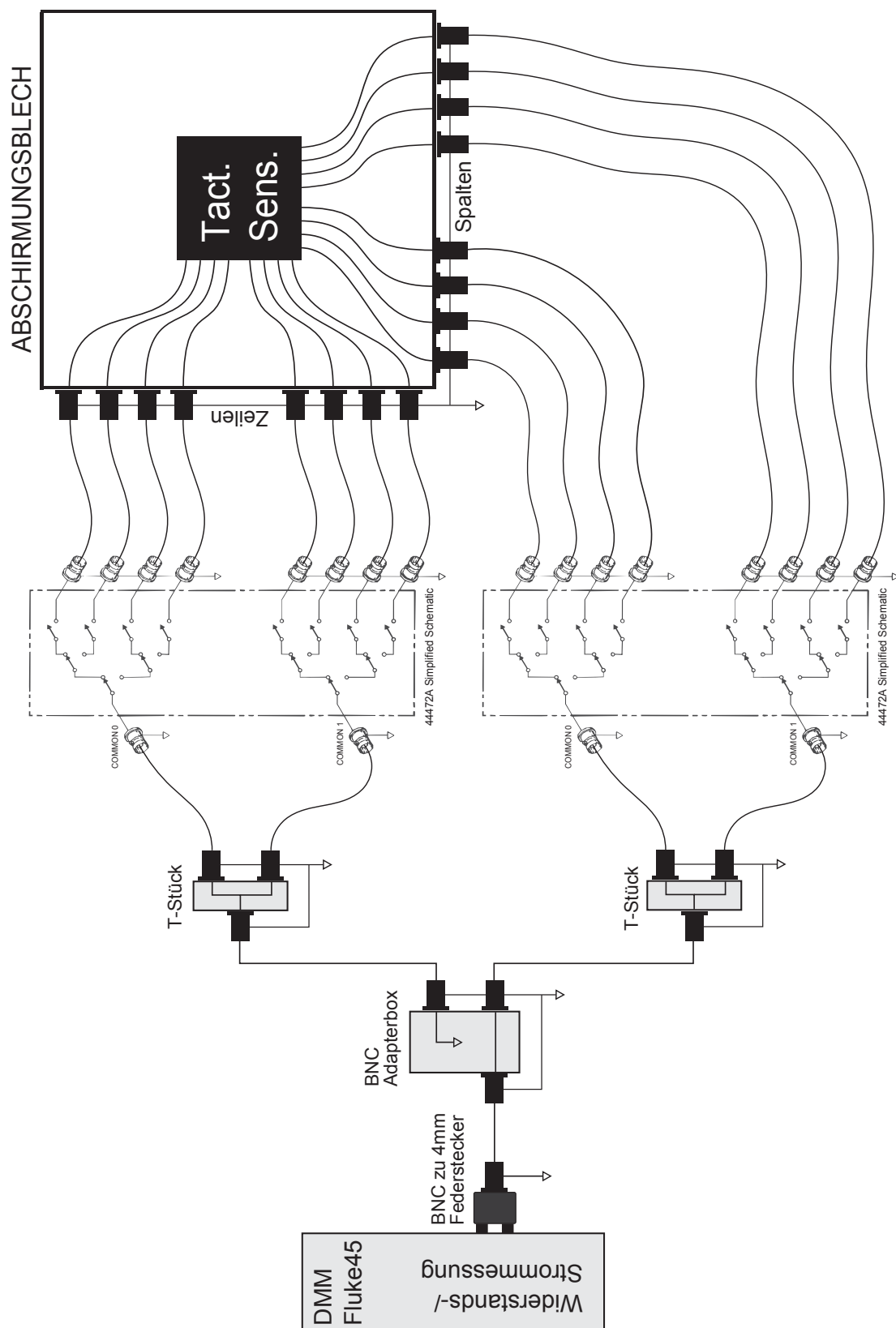


Abbildung 3.16: EMF BNC Schaltplan

### 3.2.4 Evaluierung möglicher Sensorschnittstellen

Als Grundlage für einen standardisierten Test verschiedener Arten taktiler Sensoren ist es notwendig, ein breites Spektrum an Datenbussen anzubieten. Die nachfolgende Auflistung ist eine Zusammenstellung der meist verwendeten Schnittstellen für den Datenaustausch.

- |                |                |                |
|----------------|----------------|----------------|
| • Parallelport | • CAN-Bus      | • Ethernet     |
| • RS232        | • FlexRay      | • USB          |
| • RS485        | • Modbus       | • Firewire     |
| • GPIB         | • AS-Interface | • Bluetooth    |
| • I2C / TWI    | • Profibus     | • Wireless LAN |
| • LIN-Bus      | • Interbus     | • NFC          |
| • SPI          | • EtherCAT     |                |

Weiterführende Informationen zu Bussystemen sind den Quellen [1] und [20] zu entnehmen.

Da die Implementierung aller Schnittstellen den Rahmen dieser Arbeit überschreiten würde, werden nur die für das Institut notwendigen Anschlüsse verwendet.

## 4 Software

### 4.1 Steuerungssoftware Mehrachslinearsystem

Die Schrittmotortreiberkarte SMC-TR-1000 der EMIS GmbH erlaubt die Steuerung von drei Schrittmotoren über den Schritt-Richtungs-Modus. Das heißt, für jeden der drei Schrittmotoren steht ein TTL Eingang für ein Schrittsignal und ein Richtungssignal zur Verfügung. Um eine komfortable und einfache Steuerung des Mehrachslinearsystems zu ermöglichen, wird zur Ansteuerung der Schrittmotortreiberkarte ein Mikrocontroller-Evaluierungsboard STK500 der Atmel Corporation eingesetzt. Ziel ist es, über dieses Evaluierungsboard eine Schnittstelle zwischen Schrittmotorkarte und einem Personal Computer zu schaffen, wobei der PC lediglich zur Benutzereingabe von Positions-, Richtungs- und Geschwindigkeitssteuerbefehlen dienen soll. Eine weitere Motivation bestand darin, den Echtzeitsteuerungsteil aus dem nur bedingt echtzeitfähigen Betriebssystem des PCs auszulagern.

Die Wahl der Schnittstelle zwischen Mikrocontroller-Evaluierungsboard und Personal Computer fiel auf die serielle Schnittstelle, da diese über einen Pegelwandler am STK500 schon vorhanden war und des Weiteren noch an vielen Arbeitsrechnern vorzufinden ist oder einfach über USB-zu-Seriell Adapter nachgerüstet werden kann.

Nachfolgend finden sich die wesentlichen Abschnitte des in C verwirklichten Steuerprogramms, welches sich auf einem Atmega16 auf dem STK500 befindet, als Ablaufdiagramm dargestellt, siehe Abb. 4.1. Der kommentierte Quellcode ist unter Quellcode C.1 aufgeführt.

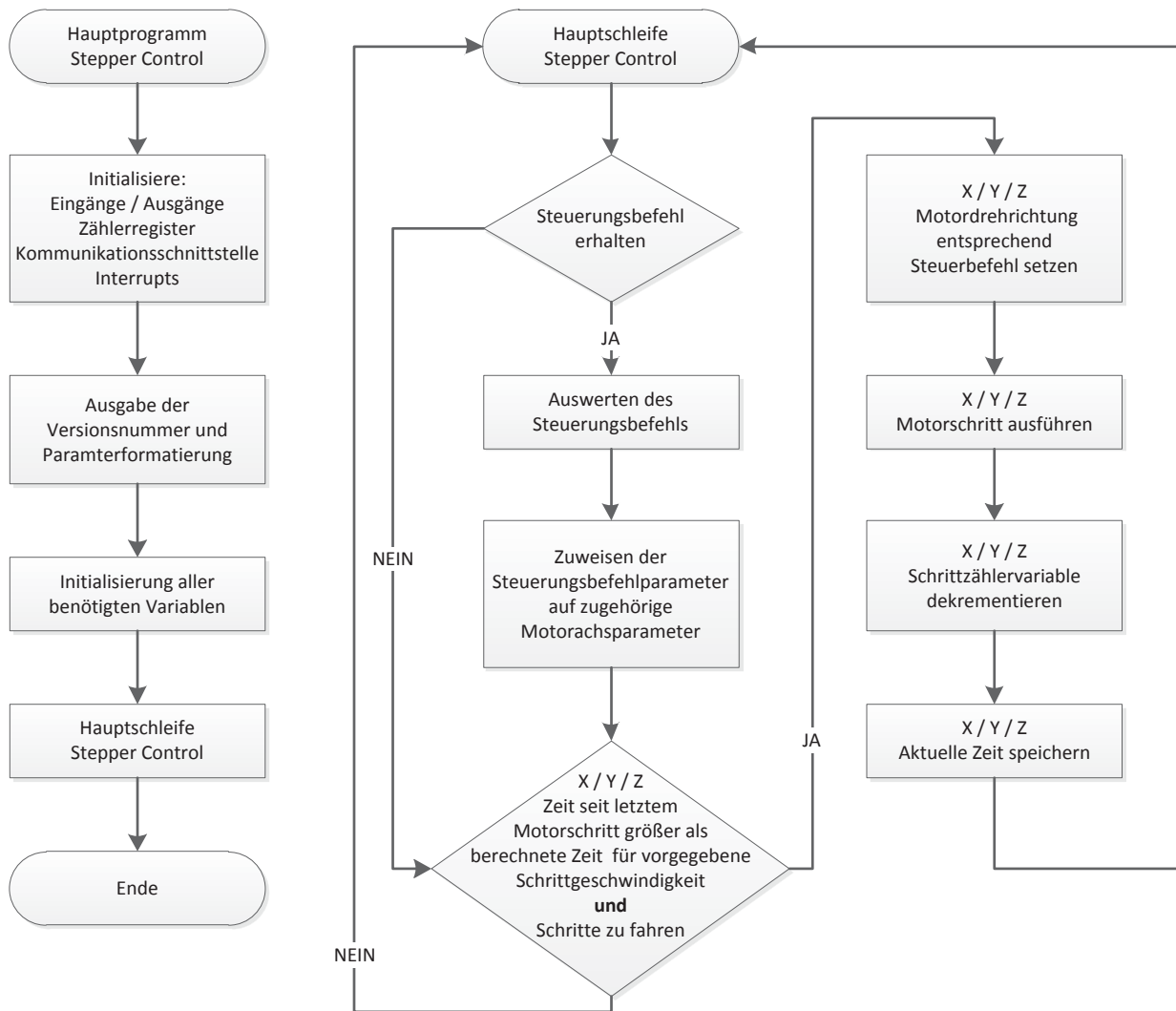


Abbildung 4.1: Ablaufdiagramm "Stepper Control" Hauptschleife

## 4.2 Auswahl einer geeigneten Messsoftware

### Auswahlkriterien

- Kompatibilität mit vorhandenem GPIB USB-Interface
- Schnelle und einfache Generierung einer Benutzeroberfläche
- Einfache Erweiterbarkeit der Benutzeroberfläche
- Weitestgehend fertige Kommunikationsschnittstellen
- Schnelle und flexible Auswertung und Weiterverarbeitung der Messergebnisse
- Selbständige Archivierung der ausgewerteten Messdaten in mehreren Dateiformaten
- Verwendung vorhandener Softwarelösungen

Aufgrund dieser Kriterien wird für die Bedienung, Visualisierung und Messwerterfassung des Prüfstandes die grafische Programmieroberfläche LabVIEW<sup>®</sup> verwendet. Für die Aufbereitung und grafische Auswertung der Messdaten fiel die Entscheidung auf MATLAB<sup>®</sup>. Die Kombination der beiden Softwarelösungen deckt alle angeführten Anforderungen ab.

## 4.3 Prüfvorschriften

### 4.3.1 Prüfarten

Neben der statischen Prüfung einzelner Taxeelemente taktiler Sensoren (siehe Abb. 4.2) werden weitere Materialprüfverfahren verwirklicht. Um eine Vorauswahl geeigneter Sensorbahnmaterialeien ohne zeit- und kostenintensive Prototypenfertigung treffen zu können, ist ein weiterer Zug- und Druckversuch von Bedeutung. Im Zugversuch einzelner taktiler Sensorbahnen wird eine Bahn ähnlich einem Schulterstab an beiden Enden eingespannt und es werden Messklemmen zur Widerstandsmessung angebracht, um das Verhalten von Kraft, Weg/Dehnung und Widerstand zu untersuchen, siehe Abb. 4.2. Der Druckversuch zeigt das Verhalten der Kontaktfläche zwischen zwei Sensorbahnen in Bezug auf Kraft, Weg und Widerstand, siehe Abb. 4.2.

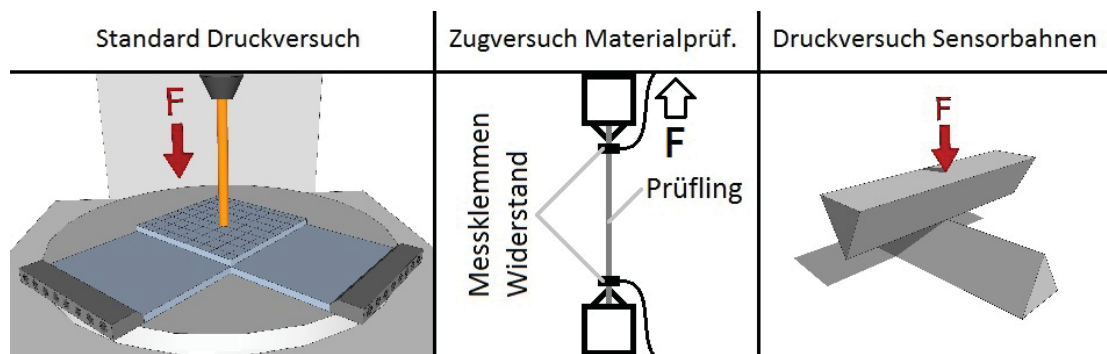


Abbildung 4.2: Prüfvorschriften taktile Sensoren und Materialeigenschaften

### 4.3.2 Implementierung der Prüfvorschriften

Vorwort:

LabVIEW<sup>®</sup> Programme bestehen im Grundsatz aus einer Benutzeroberfläche und einem dazugehörigen Blockdiagramm. Des Weiteren wird es in sogenannte Hauptprogramme namens VI ( = virtual instrument) und Unterprogramme namens SubVI unterteilt. Jedes Programm kann mehrere Schleifen, Fallunterscheidungen und Sequenzen beinhalten. Für SubVIs müssen eigens Symbolblöcke und Verbindungen angelegt werden, mit denen sie im Hauptprogramm eingebunden werden können.

Im folgenden Abschnitt ist eine Kurzübersicht über die Hauptelemente aus LabVIEW<sup>®</sup> zu finden:








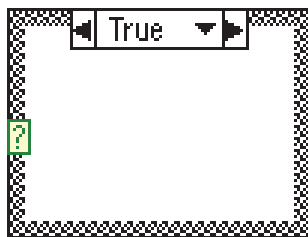
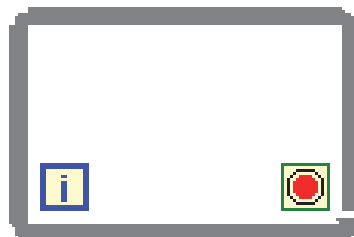
Leitungsfarben = Datentyp		
Grün		Boolean
Blau		Integer
Orange		Floating Point
Pink		String
Lila		Visa resource name out
Türkis		Dateiname/Dateipfad
Beige		Errorcode cluster

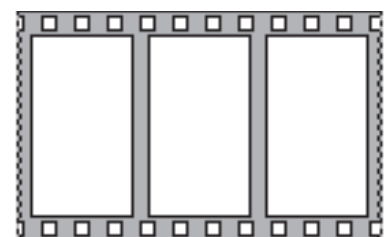
Abbildung 4.3: LabVIEW® Blockschartplan Leitungsfarben/Datentypen



(a) Fallunterscheidung



(b) While Schleife



(c) Sequenz

Abbildung 4.4: LabVIEW® Blockschartplan Strukturen

In Abb. 4.4 sind die drei Hauptstrukturen des LabVIEW® Blockschartplans aufgelistet. Ihre Funktionsweisen sind equivalent zu typischen Programmiersprachen wie zum Beispiel C++.

Nachfolgend werden kleine Ausschnitte oft verwendeter Schaltplanblöcke erläutert:

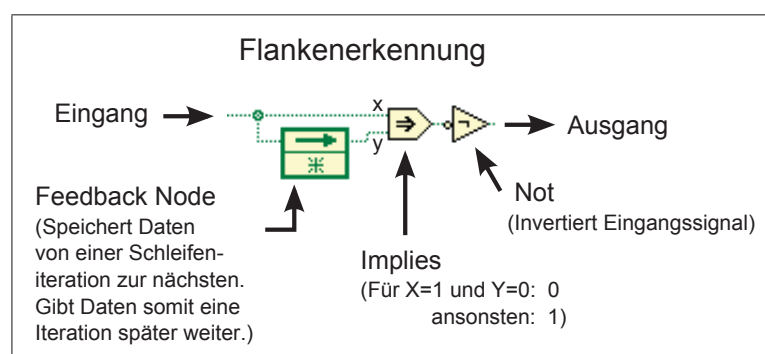


Abbildung 4.5: LabVIEW® Blockschartplan Flankenerkennung

Die in Abb. 4.5 dargestellte Flankenerkennung ermöglicht das einmalige Auslösen von Ereignissen beim Auftreten einer positiven Flanke und wird in den nachfolgenden Blockschartplänen des öfteren verwendet.

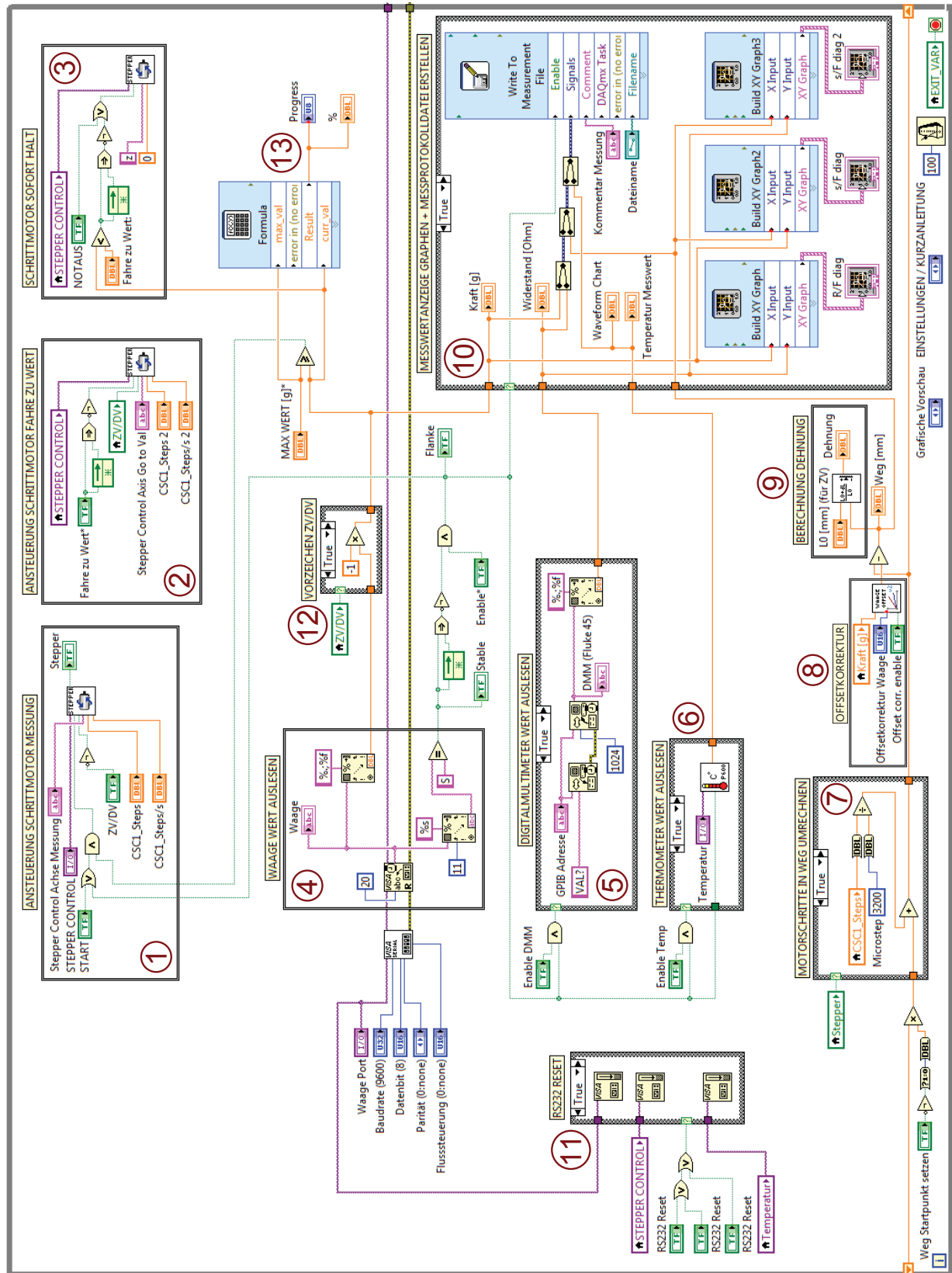


Abbildung 4.6: LabVIEW® Blocksaltplan Übersicht Hauptschleife



Die Hauptschleife des LabVIEW®-Blockschaltplans setzt sich im wesentlichen aus folgenden 13 Komponenten (siehe Abb. 4.6) zusammen:

1. Ansteuerung des Schrittmotors während des Messvorgangs
2. Ansteuerung des Schrittmotors für Funktion „Fahre zu Wert“
3. Schrittmotor Sofort-Halt
4. Messwert der Waage auslesen
5. Messwert des Digitalmultimeters auslesen
6. Messwert des Digitalthermometers auslesen
7. Motorschrittbefehle in realen Weg umrechnen
8. Wegfehler Waage Offsetkorrektur
9. Berechnung der Dehnung für Zugversuch
10. Grafische Anzeige der Messwerte und erstellen des Messprotokolls
11. Rücksetzen der seriellen Schnittstellen im Fehlerfall
12. Anpassung des Vorzeichens für Zug-/Druckversuch
13. Berechnung und Anzeige des Messvorgangfortschritts

(Detailansichten aller Blockschaltpläne des LabVIEW®-Programms siehe Anhang E)

## 4.4 Benutzeroberfläche des Prüfstandes

Die mit LabVIEW<sup>®</sup> erstellte Benutzeroberfläche zur Bedienung des Prüfstandes ist in fünf Hauptbereiche (siehe Abb. 4.7) unterteilt:

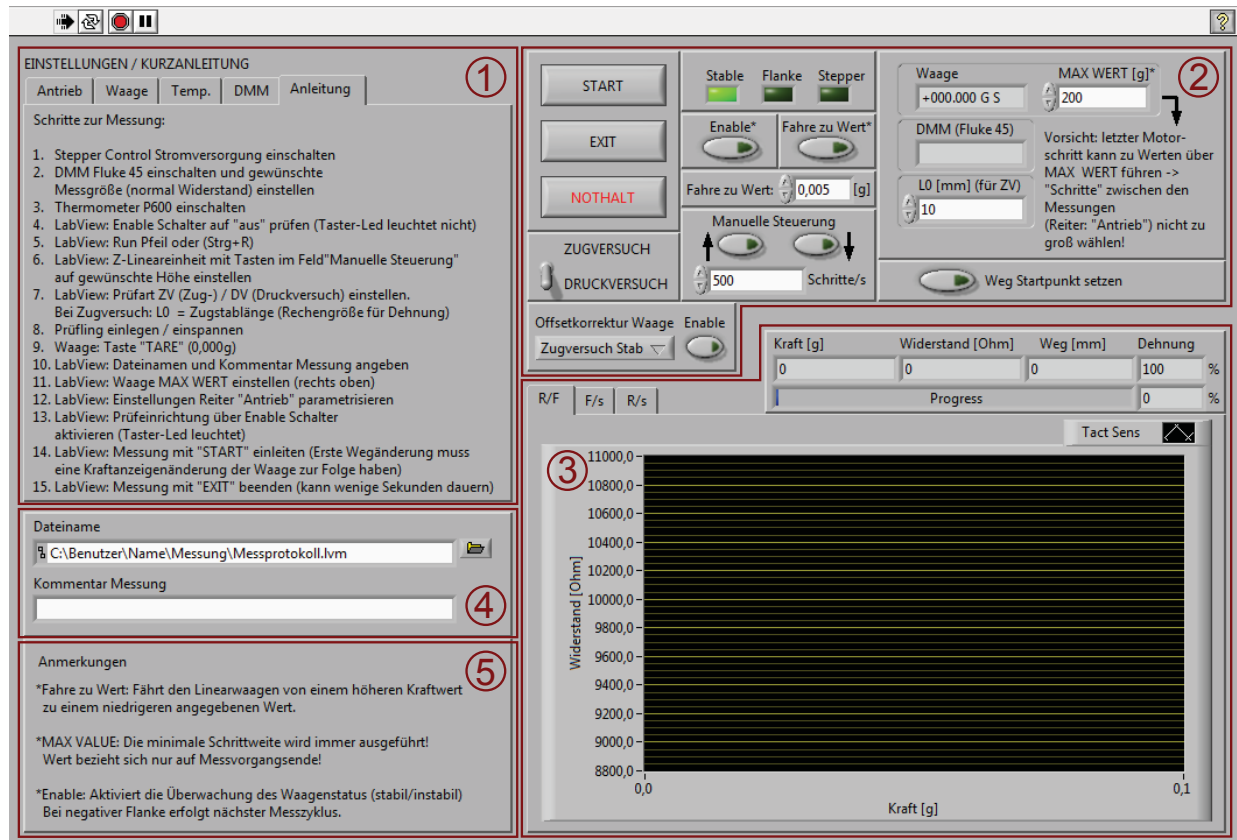
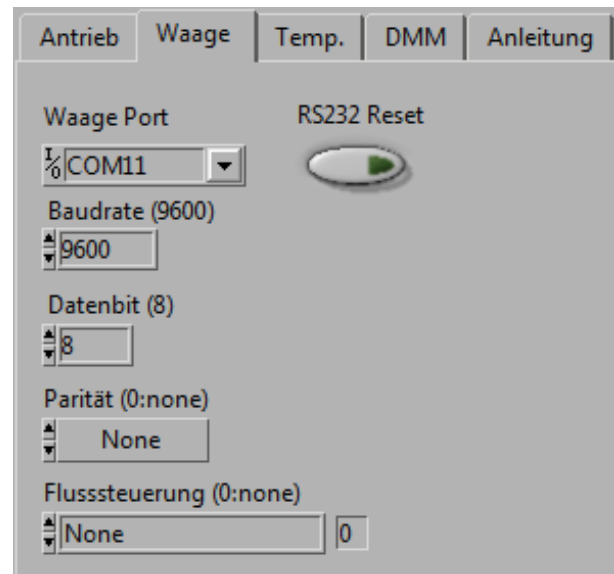


Abbildung 4.7: LabVIEW<sup>®</sup> Startbildschirm Benutzeroberfläche Prüfstand

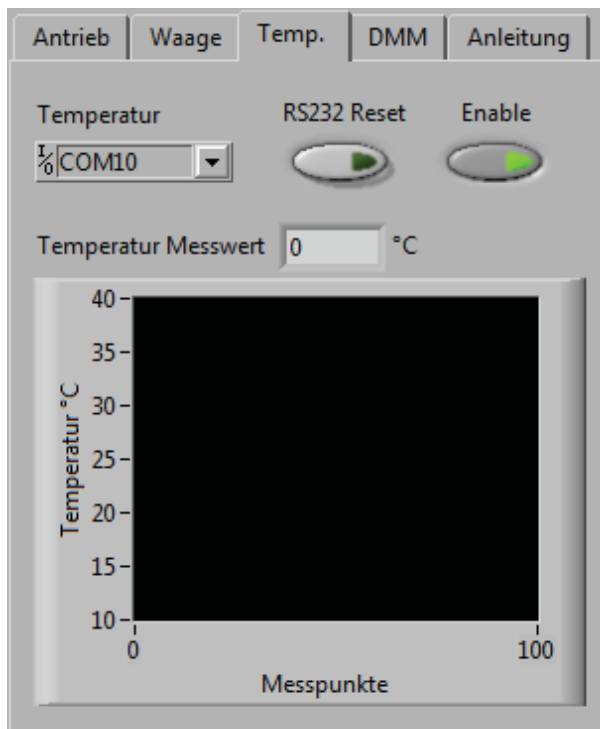
1. Einstellungen für Antrieb und Messgeräte und Kurzanleitung
2. Bedienelemente zur Steuerung des Prüfstandes
3. Grafische Anzeige und Auswertung der laufenden Messung
4. Ausgabe des Messprotokolls (Speicherort und Kommentar)
5. Zusätzliche Anmerkungen zu Bedienelementen



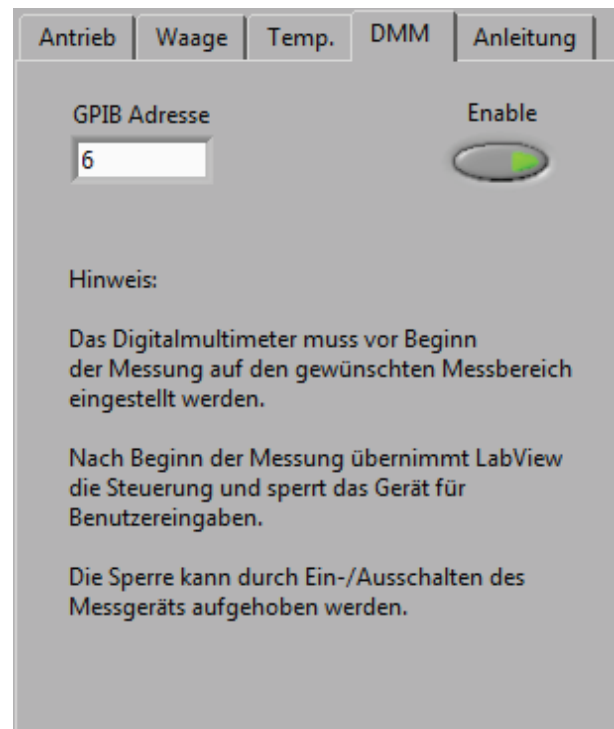
(a) Antrieb/Vorschub



(b) Waage/Kraftmessung



(c) Thermometer/Temperaturerfassung



(d) Digitalmultimeter/Widerstandsmessung

Abbildung 4.8: LabVIEW® Benutzeroberfläche Einstellungen

#### Abb. 4.7 (1) Einstellungen für Antrieb und Messgeräte:

- Abb. 4.8 (a): Reiter Antrieb  
Hier sind alle Einstellungen für die Hublineareinheit (Z-Achse) vorzunehmen. Im I/O Feld ist die serielle Schnittstelle der Schrittmotorsteuerkarte auszuwählen. Unter „Messablauf (Stepper Control Parameter)“ sind die logische Achse der Schrittmotortreiberstufe, die Wegauflösung der Messung in Schritten und die Geschwindigkeit in Schritten pro Sekun-

de anzugeben. In den Feldern von „Fahre zu Wert (Stepper Control Parameter)“ kann ebenfalls wieder die Achse, der maximal zu fahrende Weg in Schritten und die Geschwindigkeit in Schritten pro Sekunde angegeben werden. (Die Funktion von „Fahre zu Wert“ ist unter 2. Bedienelemente beschrieben). Die Taste „RS232 Reset“ verwirft den Pufferspeicher der seriellen Schnittstelle. Dies kann erforderlich sein, wenn das Prüfprogramm nicht mit der „EXIT“ Taste beendet wurde.

- Abb. 4.8 (b): Reiter Waage

Hier sind alle Einstellungen für die serielle Kommunikation mit der Waage auszuwählen. Da die verwendete Waage KERN EW EG420-3NM verschiedene Baudraten und sonstige Einstellungen zulässt, können diese Parameter ebenfalls angepasst werden. Der „RS232 Reset“ Taster ist im vorhergehenden Reiter Antrieb erklärt.

- Abb. 4.8 (c): Reiter Temp.

Hier ist die serielle Schnittstelle an dem das Thermometer (Dostmann P600) angeschlossen ist auszuwählen. Der „RS232 Reset“ Taster ist im vorhergehenden Reiter Antrieb erklärt. Der „Enable“ Taster erlaubt die Temperaturerfassung zu deaktivieren, für den Fall, dass kein Thermometer vorhanden sein sollte. Des Weiteren beinhaltet dieser Reiter eine grafische Anzeige über die letzten 100 Temperaturmesspunkte, die während der laufenden Prüfung aufgenommen wurden. Im Feld „Temperatur Messwert“ kann der letzte Messpunkt direkt abgelesen werden.

- Abb. 4.8 (d): Reiter DMM

Das verwendete Digitalmultimeter (Fluke 45) wird über die GPIB Schnittstelle ausgelesen. Dazu muss im Feld „GPIB Adresse“ die eingestellte Adresse des Messgerätes eingetragen werden. Der „Enable“ Taster erlaubt die Messwertauslesung des Digitalmultimeters zu deaktivieren.

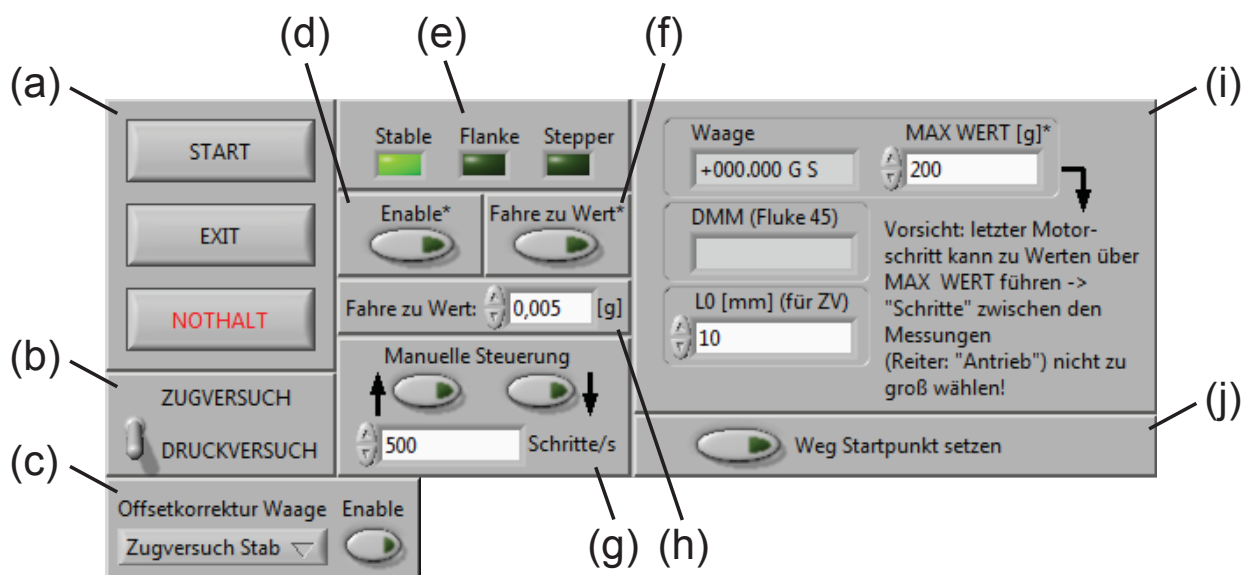


Abbildung 4.9: LabVIEW® Benutzeroberfläche Bedienelemente Steuerung

**Abb. 4.7 (2) Bedienelemente zur Steuerung des Prüfstandes:**

- Abb. 4.9 (a): Hauptsteuerelemente  
START: Verfährt die Hublineareinheit einen Schritt um die eingestellte Wegauflösung in Prüfrichtung. Dies sollte eine messbare Kraftänderung auf der Waage zur Folge haben, welche wiederum ihren Status auf „Messwert instabil“ ändert (siehe Abb. 4.9 (e) Indikator Stable leuchtet nicht). Falls der Taster „Enable“ aus Abb. 4.9 (d) gesetzt ist, erfolgt der nächste Hubschritt in Prüfrichtung, sobald die Waage wieder stabil anzeigt (siehe Abb. 4.9 (e) Indikator Stable leuchtet). Die Messung läuft nun stetig weiter.  
  
EXIT: Beendet das laufende LabVIEW® Programm des Prüfstandes und gibt alle Steuer- und Messschnittstellen frei.  
  
NOTHALT: Sendet sofort einen Stopbefehl an die „Stepper Control“ der Hublineareinheit.
- Abb. 4.9 (b): Prüffart  
Über die Schalterstellung wird zwischen Druckversuch und Zugversuch gewählt. Diese Option legt das Vorzeichen der Kraftmessung fest und bestimmt die Fahrtrichtung der Hublineareinheit in Prüfrichtung. Das heißt, bei einem Zugversuch fährt die Hublineareinheit nach oben und bei einem Druckversuch nach unten.
- Abb. 4.9 (c): Offsetkorrektur Waage  
Sie bezieht sich auf den Wegfehler der Waage, der sich durch die aufgebrachte Kraft ergibt. Über das Dropdown-Menü kann für die drei verfügbaren Prüffarten (Zugversuch Stab, Druckversuch Taxel, Druckversuch Bahnen) die entsprechende Korrekturkurve ausgewählt werden. Mit dem Taster „Enable“ wird die Offsetkorrektur aktiviert.
- Abb. 4.9 (d): „Enable“ Taster  
Der Taster „Enable“ aktiviert die Überwachung der Statusanzeige der Waage. Sobald ein Flankenwechsel des Waagenstatus von instabil zu stabil erfolgt, werden alle aktuellen Messwerte aufgezeichnet und anschließend die Hublineareinheit in Prüfrichtung um die angegebene Wegauflösung verfahren.
- Abb. 4.9 (e): LED Indikatoren  
Stable: Zeigt den Status der Waage an (stabil  $\Rightarrow$  leuchtet, instabil  $\Rightarrow$  leuchtet nicht)  
  
Flanke: Leuchtet bei einer negativen Flankenerkennung des Waagenstatus auf.  
  
Stepper: Leuchtet nach Ausführung eines Motorschritts der Hublineareinheit während des Messablaufs auf.
- Abb. 4.9 (f): „Fahre zu Wert“ Taster  
Ermöglicht am Ende einer Messung das Rückkehren der Hublineareinheit zu einer vorgegebenen Kraft (siehe 4.9 (h)).
- Abb. 4.9 (g): Manuelles Verfahren der Hublineareinheit  
Im Texteingabefeld kann eine Verfahrensgeschwindigkeit in Schritten pro Sekunde angegeben werden, mit der die Hublineareinheit manuell über die beiden Taster nach oben oder nach unten, entsprechend der Pfeilrichtungen, verfahren werden kann. Dies ist meist am Anfang einer Messung erforderlich um z.B. bei einem Zugversuch die Einspannklemmen auf Einspannlänge des Prüflings zusammenzufahren.

- Abb. 4.9 (h): „Fahre zu Wert:“ Eingabefeld  
Hier ist die Eingabe des anzufahrenden Kraftwertes zu Abb. 4.9 (f) erforderlich.
- Abb. 4.9 (i): Ausgelesene Zeichenfolge Waage, DMM; Abbruchbedingung; Einspannlänge  
Anzeigefeld Waage: Zeigt den unbearbeiteten aktuellen Messwert der Digitalwaage die Messwerteinheit (G = Gramm) und den Status der Messung (S = Messwert stabil, U = Messwert instabil) an.  
  
Eingabefeld MAX WERT: Es ist der Maximalwert für die Messwerte der Waage anzugeben, bei welchem die Messung angehalten werden soll.  
  
Anzeigefeld DMM: Zeigt den unbearbeiteten ausgelesenen Wert des Digitalmultimeters an.  
  
Eingabefeld L0: Hier kann im Falle eines Zugversuches die Zugstablänge angegeben werden. Der Wert dient lediglich einer prozentualen Anzeige der Dehnung im Benutzeroberflächenbereich 3.
- Abb. 4.9 (j): Weg Startpunkt setzen  
Der Taster setzt den Wegzähler für die aktuelle Position der Hublineareinheit auf Null. Dies kann am Anfang einer Messung erforderlich sein, falls z.B. eine vorhergehende Messung abgebrochen wurde.

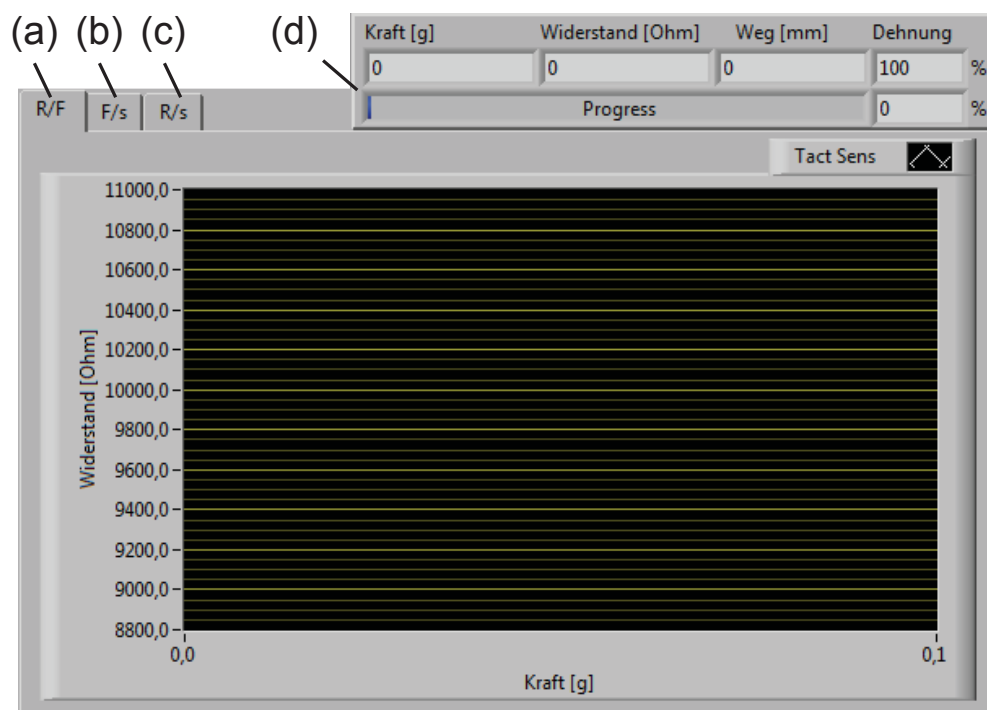


Abbildung 4.10: LabVIEW® Benutzeroberfläche grafische Anzeige und Auswertung

### Abb. 4.7 (3) Grafische Anzeige und Auswertung der laufenden Messung:

- Abb. 4.10 (a): Grafik: Elektrischer Widerstand zu aufgebrachter Kraft  
Stellt die bisher aufgezeichneten Messpunkte in einem X-Y-Diagramm dar.

- Abb. 4.10 (b): Grafik: Aufgebrachte Kraft zu verfahrenem Weg  
Stellt die bisher aufgezeichneten Messpunkte in einem X-Y-Diagramm dar.
- Abb. 4.10 (c): Grafik: Elektrischer Widerstand zu verfahrenem Weg  
Stellt die bisher aufgezeichneten Messpunkte in einem X-Y-Diagramm dar.
- Abb. 4.10 (d): Messwerte und Fortschrittsanzeige  
Die oberen drei Textfelder Kraft, Widerstand und Weg zeigen die aktuellen Messwerte des letzten Messpunktes. Das Feld Dehnung ist nur im Falle eines Zugversuches relevant und bezieht sich auf die aktuelle Längenänderung im Verhältnis zur angegebenen Einspannlänge  $L_0$ , siehe Abb. 4.9 (i). In der unteren Zeile wird der aktuelle Messfortschritt grafisch und numerisch angezeigt. Er bezieht sich auf den aktuellen Messwert der Waage im Verhältnis zu dem eingetragenen Wert im Feld „MAX WERT“, siehe Abb. 4.9 (i).

**Abb. 4.7 (4) Ausgabe des Messprotokolls (Speicherort und Kommentar):**

Im Eingabefeld „Dateiname“ ist der Speicherort des von LabVIEW<sup>®</sup> erzeugten Messprotokolls zu wählen.

Im Eingabefeld „Kommentar Messung“ können zusätzliche Angaben zu dem zu prüfenden Material gemacht werden. Diese werden im Messprotokoll mit abgelegt.

**Abb. 4.7 (5) Zusätzliche Anmerkungen zu Bedienelementen:**

Hier sind die Ausführungen der Fußnoten einzelner Bedienelemente zu finden.

## 4.5 Verarbeitung und Aufbereitung der Messdatenprotokolle

Nachfolgend wird in Abb. 4.11 die Weiterverarbeitung der mit LabVIEW® erstellten Messdatenprotokolle mithilfe der Software MATLAB® erläutert. Der vollständig kommentierte Programmcode ist im Anhang unter Quellcode D.1 aufgeführt.

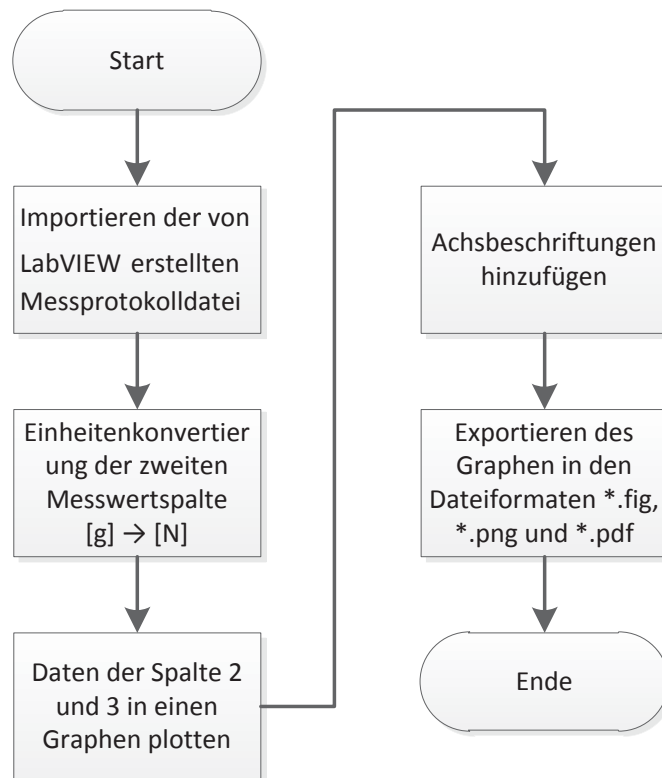


Abbildung 4.11: Ablaufdiagramm Messdatenverarbeitung mit MATLAB®

Es wird die Messdatendatei mit MATLAB® geöffnet. Die ersten 23 Zeilen sind der von LabVIEW® erzeugte Kopf des Messprotokolls. Die Messdaten beginnen ab Zeile 24, werden Spaltenweise aufgelistet und sind untereinander durch Tabulatoren getrennt. MATLAB® liest diese Daten ein und legt sie in einer neuangelegten zweidimensionalen Matrix, zur weiteren Bearbeitung, ab. Anschließend erfolgt die Umrechnung der Messdaten der Feinwaage, da diese in der Einheit Gramm aufgenommen werden und die Einheit Newton gewünscht ist. Zu diesem Zweck werden die Messwerte der zweiten Messspalte mit dem Faktor  $(9,81 \frac{m}{s^2} \div 1000) = 0,00981 \frac{m}{s^2}$  multipliziert.

Die Daten liegen nun im richtigen Format vor und können grafisch ausgewertet werden. Dazu wird in MATLAB® mit dem Befehl „plot“ und der Angabe der gewünschten Messspalten für die Daten der X und Y-Achse eine Grafik erzeugt. Desweiteren werden die Achsbeschriftungen hinzugefügt und optional eine Legende erstellt.

Zuletzt werden die fertigen Grafiken in den gängigsten Dateiformaten exportiert.



## 5 Messergebnisse

### 5.1 Aufnahme von Messdaten

Für alle Widerstandsmessungen die mit dem Prüfstand dieser Arbeit und folglich mit dem Digitalmultimeter Fluke 45 durchgeführt werden, sind die Genauigkeitsangaben der jeweiligen Messbereiche nach Abb. 5.1 zu beachten.

Für niederohmige, taktile Sensoren ist der Zuleitungswiderstand der Messleitungen ebenfalls zu beachten. Er kann mit der Relativ-Zusatzfunktion des Digitalmultimeters Fluke 45 abgeglichen werden.

Meßbereich	Auflösung			Genauigkeit	Typische Meßspannung am Bereichsende	Max. Strom
	niedriger	mittlerer	hoher			
300 $\Omega$	—	10 m $\Omega$	100 M $\Omega$	0.05 % + 2 + 0.02 $\Omega$	0.25	1 mA
3 k $\Omega$	—	100 M $\Omega$	1 $\Omega$	0.05 % + 2	0.24	120 $\mu$ A
30 k $\Omega$	—	1 $\Omega$	10 $\Omega$	0.05 % + 2	0.29	14 $\mu$ A
300 k $\Omega$	—	10 $\Omega$	100 $\Omega$	0.05 % + 2	0.29	1.5 $\mu$ A
3 M $\Omega$	—	100 $\Omega$	1 k $\Omega$	0.06 % + 2	0.3	150 $\mu$ A
30 M $\Omega$	—	1 k $\Omega$	10 k $\Omega$	0.25 % + 3	2.25	320 $\mu$ A
300 M $\Omega^*$	—	100 k $\Omega$	1 M $\Omega$	2 %	2.9	320 $\mu$ A
100 $\Omega$	1 m $\Omega$	—	—	0.05 % + 8 + 0.02 $\Omega$	0.09	1 mA
1000 $\Omega$	10 m $\Omega$	—	—	0.05 % + 8 + 0.02 $\Omega$	0.10	120 $\mu$ A
10 k $\Omega$	100 m $\Omega$	—	—	0.05 % + 8	0.11	14 $\mu$ A
100 k $\Omega$	1 $\Omega$	—	—	0.05 % + 8	0.11	1.5 $\mu$ A
1000 k $\Omega$	10 $\Omega$	—	—	0.06 % + 8	0.12	150 $\mu$ A
10 M $\Omega$	100 $\Omega$	—	—	0.25 % + 6	1.5	150 $\mu$ A
100 M $\Omega^*$	100 k $\Omega$	—	—	2 % + 2	2.75	320 $\mu$ A
* Bedingt durch die von diesem Meßgerät zur Widerstandsmessung verwendete Methode sind die 100 M $\Omega$ (niedrige Meßgeschwindigkeit) und 300 M $\Omega$ (mittlere und hohe Meßgeschwindigkeit) -Meßbereiche zu Messungen unterhalb von 3,125 M $\Omega$ bzw. 20 M $\Omega$ ungeeignet. "UL" erscheint in der Anzeige für Widerstände unter diesen nominellen Werten, und die Computer-Schnittstelle sendet "+E1-9".						

Abbildung 5.1: Digitalmultimeter Fluke 45 Widerstandsmessung Genauigkeit [11]

Ist bei einer Messung nicht nur die Kraft-/Widerstandskennlinie sondern auch der Weg von Bedeutung, kann dessen Genauigkeit über die in der LabVIEW<sup>®</sup>-Benutzeroberfläche implementierten Offsetkompensation (Abb. 4.9 (c)) des Waagenweges für die verschiedenen Prüfarten verbessert werden, siehe Kap. 5.2.

## 5.2 Auswertung der Messergebnisse

### 5.2.1 Offsetkompensation Feinwaage: Ermittlung der Kennlinie

Die Verwendung einer Feinwaage als Kraftmesser hat den Nachteil, dass diese bei Belastung nachgibt und somit ein Wegfehler der Messung die Folge ist. Zur Kompensation dieses Fehlers muss die Kraft-/Wegkennlinie der Waage ermittelt werden. Die einfachste Variante um diese Kennlinie zu erhalten ist anstatt eines elastischen Prüfkörpers einen nicht elastischen zu verwenden. Somit konnte am Beispiel des Zugversuches nach Abb. 6.2 (ZV) mithilfe eines festen Stück Drahtes die Kraft-/Wegkennlinie der Waage nach Abb. 5.2 ermittelt werden. Anhand der Kennlinie konnte mit MATLAB® eine Trendlinie und deren Formel als Polynom zweiten Grades erstellt werden.

Polynom zweiten Grades der Trendlinie der Wegoffsetkorrektur der Feinwaage für Zugversuche:

$$y[mm] = 2 \cdot 10^{-6} \frac{mm}{g^2} \cdot x^2 + 0,001 \frac{mm}{g} \cdot x + 0,0011mm$$

(Variable x ist der Messwert der Waage in Gramm)

Die Wegoffset-Korrekturformeln für die beiden Druckversuche nach Abb. 6.2 (DV1 und DV2) wurden entsprechend dem Zugversuch ermittelt.

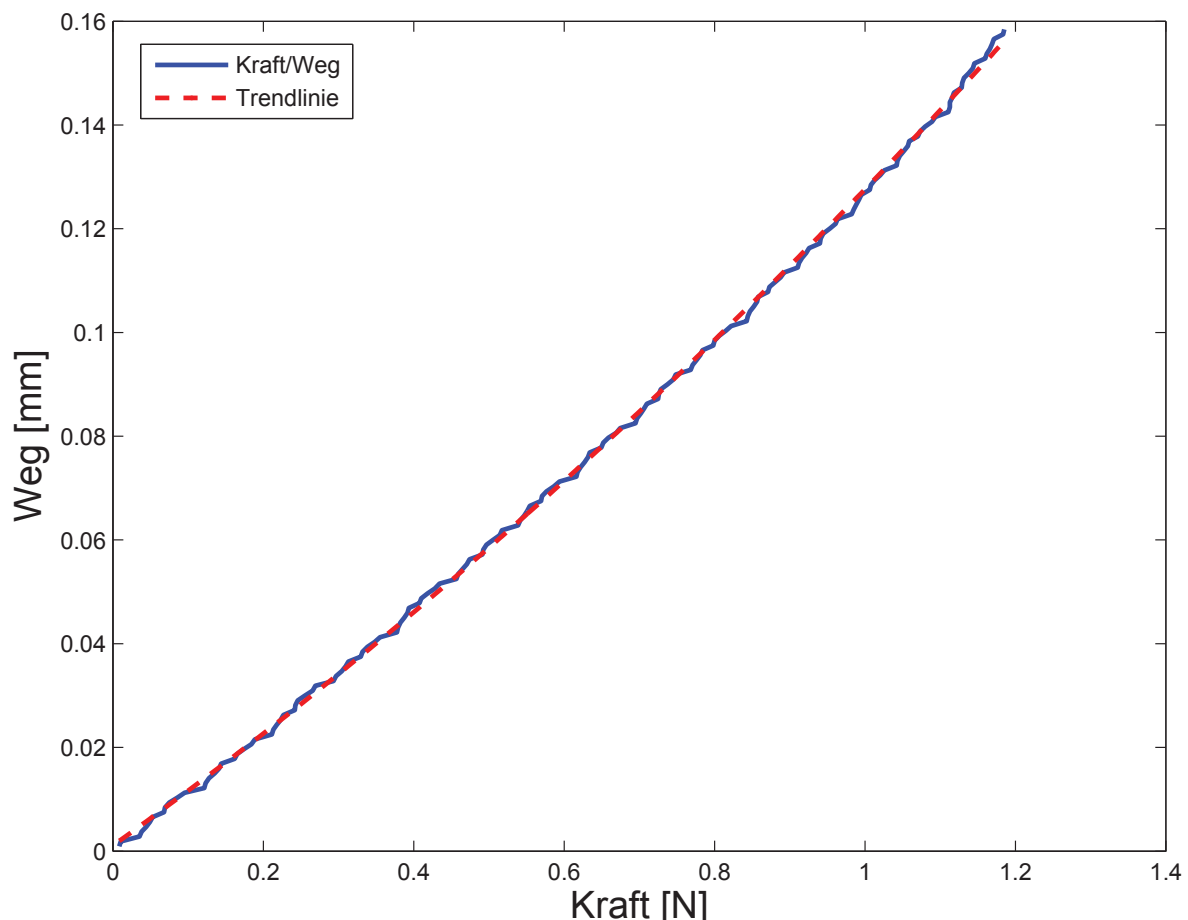


Abbildung 5.2: MATLAB® Plot Zugversuch Offsetkurve Feinwaage

### 5.2.2 Offsetkompensation Feinwaage: Überprüfung auf Funktion

Die in Kap. 5.2.1 erarbeitete Formel der Offsetwegkompensation der Feinwaage wird geprüft, indem der beschriebene Zugversuch mit einem nicht elastischen Prüfkörper mit deaktivierter und darauf mit aktivierter Offsetwegkompensation durchgeführt wird.

Einen direkten Vergleich der Kompensationswirkung zeigt die Abb. 5.3.

Die detaillierte Ansicht der Zugversuchsauswertung mit aktivierter Kompensation ist in Abb. 5.4 zu sehen.

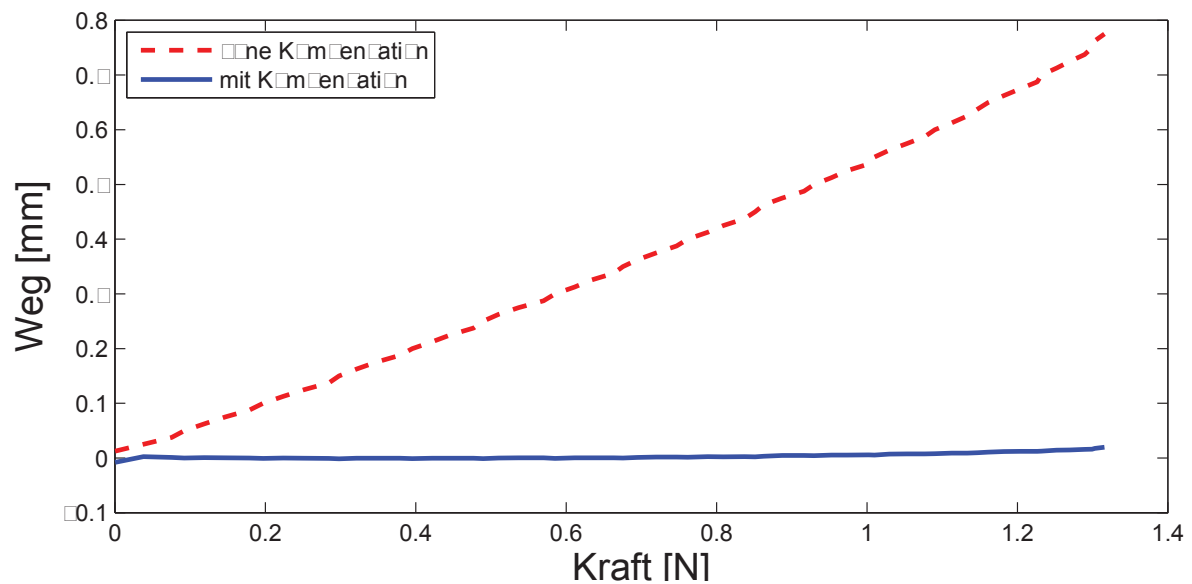


Abbildung 5.3: MATLAB® Plot Zugversuch Vergleich Wegkompensation Feinwaage

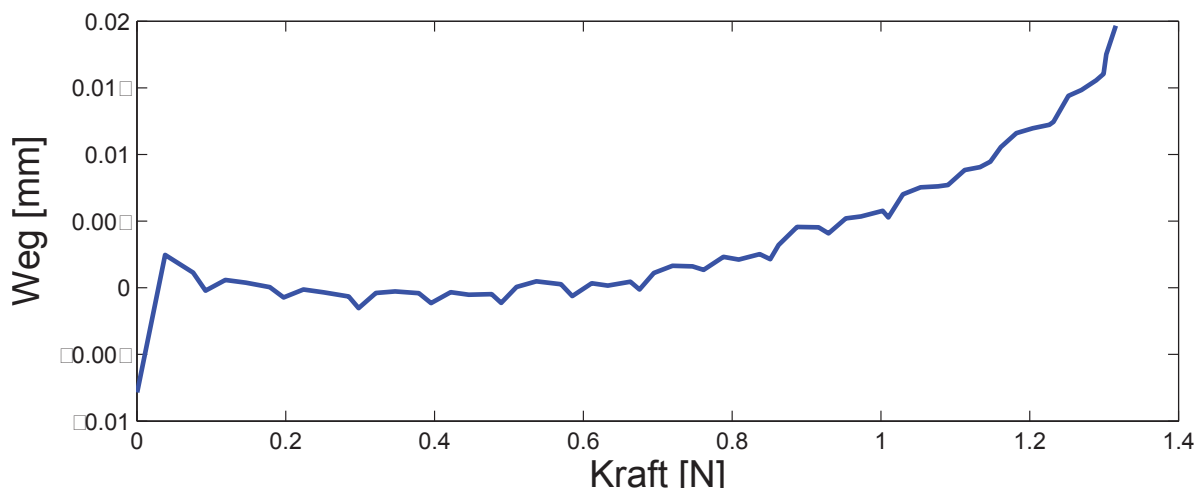


Abbildung 5.4: MATLAB® Plot Zugversuch Wegkompensation Feinwaage Detailansicht

### 5.2.3 Druckversuch von Sensorbahnen verschiedener Geometrie

Dieser Abschnitt befasst sich mit Druckversuchen von Sensorbahnen verschiedener Geometrien. Als Prüflinge liegen je zwei dreieckige Sensorbahnen (gemäß Abb. 5.5 (a)) und zwei runde Sensorbahnen (gemäß Abb. 5.5 (b)) vor. Diese werden in den Prüfstand wie in Abb. 6.2 (DV2) dargestellt eingespannt und die Prüfung durchgeführt.

Zweck dieser Prüfung ist die Eignung verschiedener Sensorbahngeometrien für neue Sensorprototypen nach [25] zu untersuchen.

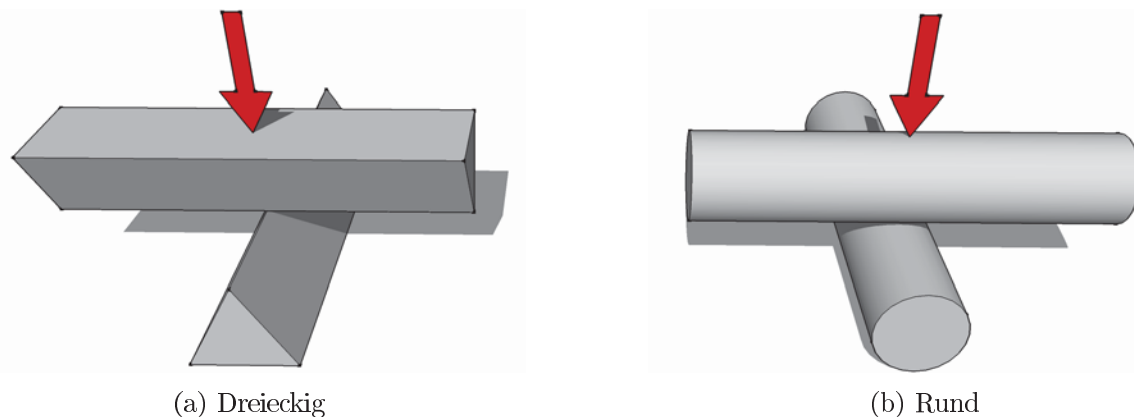


Abbildung 5.5: Skizze Druckversuch von Sensorbahnen verschiedener Geometrie

#### Geometrie: Runde Sensorbahnen

Wie in Abb. 5.6 zu sehen ist verläuft der Graph am Anfang in einem geringen Kraftbereich mit großer negativer Steigung bis auf ein stetiges Minimum, auf dem er bis auf kleinere Abweichungen verweilt.

Diese Charakteristik ist für eine taktile Sensorbahn sehr ungeeignet, da durch das annähernd binäre Verhalten nur geringe Sensorempfindlichkeiten erzielt werden könnten.

#### Geometrie: Dreieckige Sensorbahnen

Wie in Abb. 5.7 zu sehen ist verläuft der Graph am Anfang in einem geringen Kraftbereich mit großer negativer Steigung, nähert sich danach jedoch asymptotisch seinem Minimum an.

Diese Charakteristik ist für eine taktile Sensorbahn wesentlich besser geeignet, da in einem weiten Kraftbereich mehrere Widerstandswerte zugeordnet werden können.

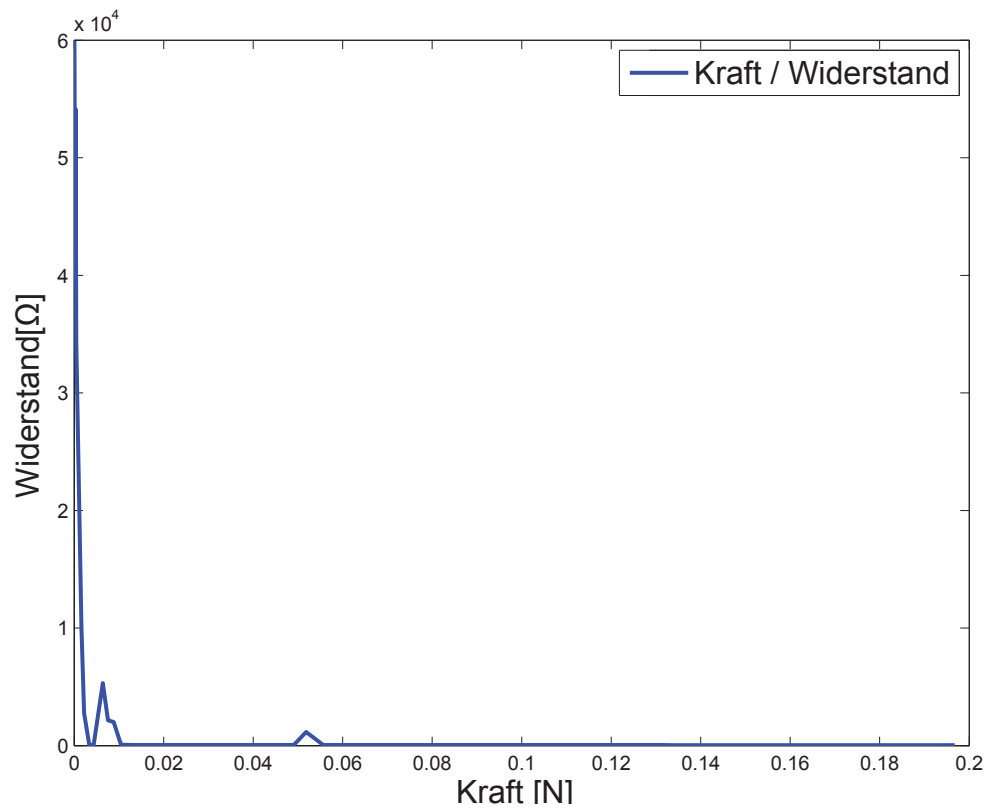


Abbildung 5.6: MATLAB<sup>®</sup> Plot Druckversuch taktile Sensorbahn rund

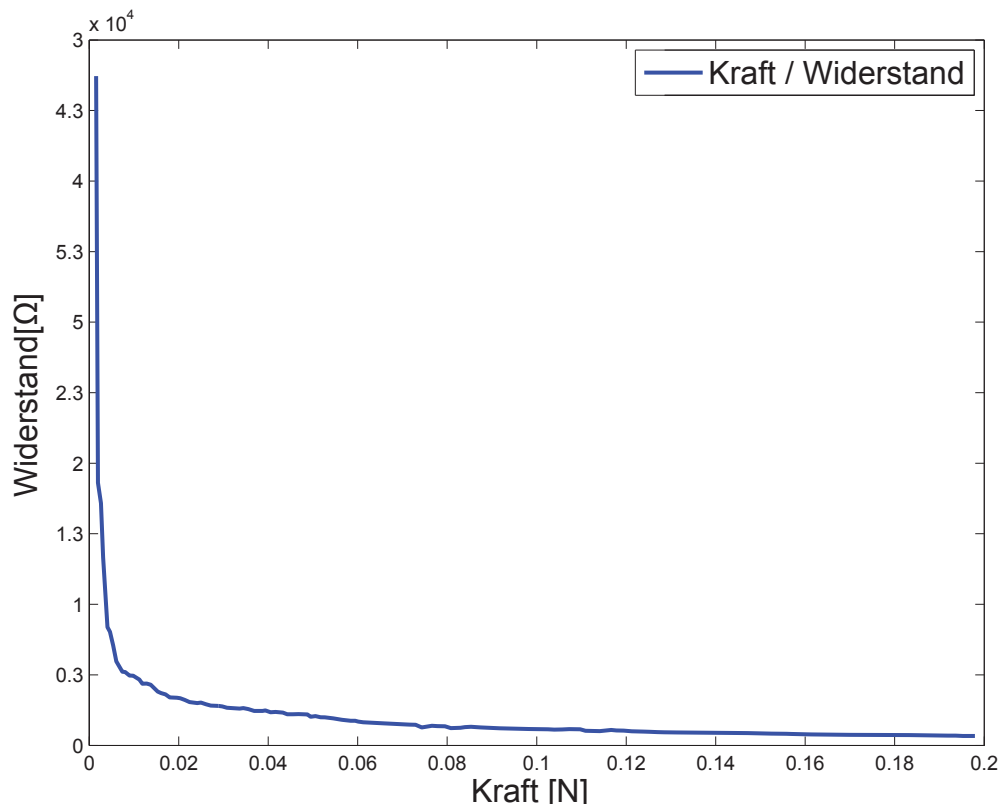


Abbildung 5.7: MATLAB<sup>®</sup> Plot Druckversuch taktile Sensorbahn dreieckig

## 6 Ergebnis

Dieses Kapitel befasst sich mit dem aus Kap. 3.1.5 realisierten Testaufbau. Wie in Abb. 6.1 dargestellt, besteht dieser aus einer kleinen Lineareinheit mit angeschraubter Adapterplatte, Halteblöcken und Schrittmotor, einem Präzisions-Temperaturmessgerät mit Temperaturfühler und einer Präzisions-Feinwaage. Der elektrische Widerstand der Sensorbahnen wird über das Digitalmultimeter „Fluke 45“ erfasst. Der Schrittmotor der Lineareinheit wird über die Schrittmortreiberkarte „SMC-TR-1000“ und dem mit einem Mikrocontroller bestückten Entwicklungskit „AVR STK 500“ angesteuert, siehe Kap. 3.2.1 und 4.1.

Die Kommunikation von Feinwaage, Temperaturmessgerät und Lineareinheit mit dem Steuerrechner erfolgt über die serielle Schnittstelle „RS232“. Das Digitalmultimeter verfügt zusätzlich zu der seriellen Schnittstelle über den leistungsfähigeren GPIB-Bus und wird daher über diesen angesteuert.

Die in LabVIEW® entwickelte Prüfsoftware erlaubt mit diesem Aufbau drei verschiedene, automatisierte, Prüfverfahren:

- Zugversuch von einzelnen Sensorbahnen (siehe Abb. 6.2 ZV)
- Druckversuch von einzelnen Messtaxeln (siehe Abb. 6.2 DV1)
- Druckversuch zweier Sensorbahnen (siehe Abb. 6.2 DV2)

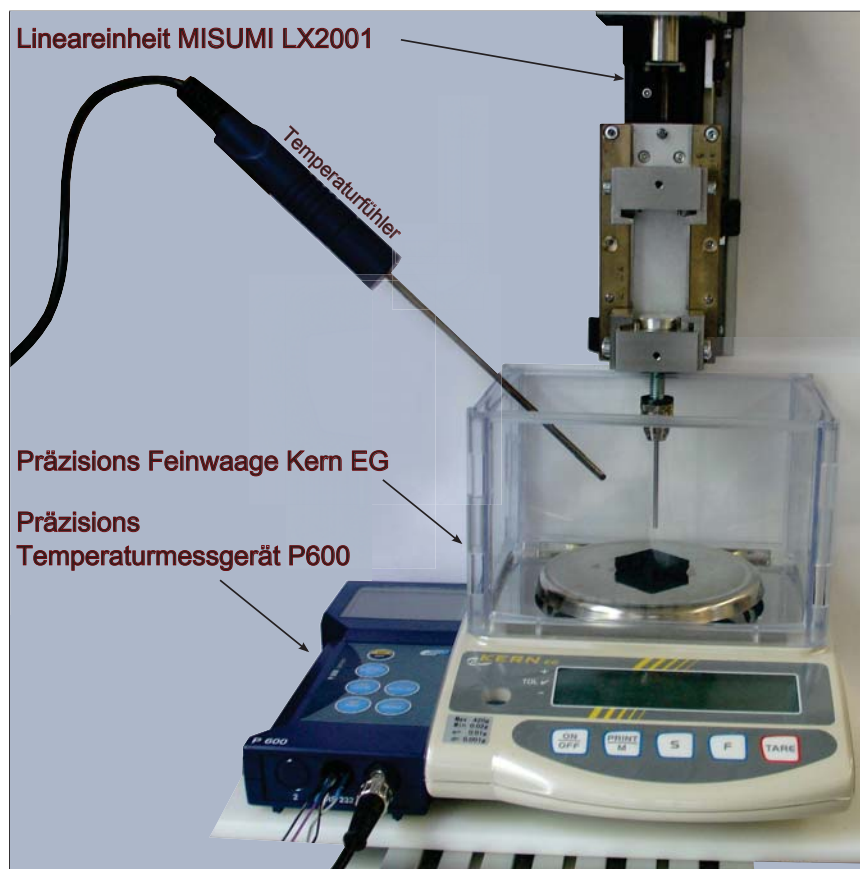


Abbildung 6.1: Testaufbau Übersicht

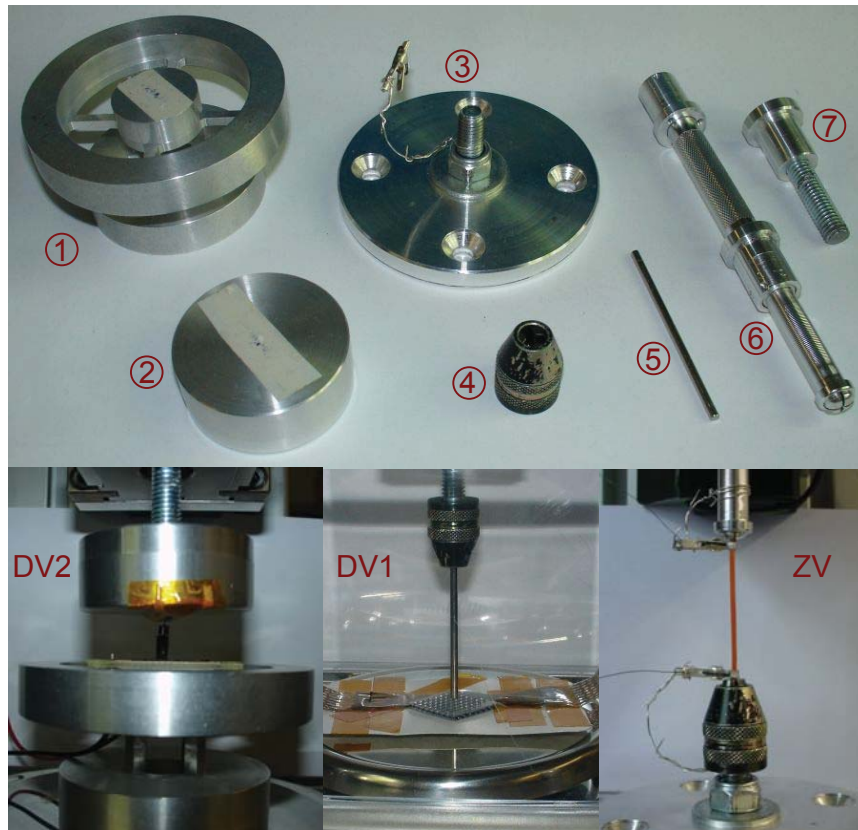


Abbildung 6.2: Prüfhalterungen und ihre Anwendung

Für die drei Prüfverfahren werden verschiedene Halterungen für die unterschiedlichen Prüflinge benötigt. Die Einzelkomponenten dieser Halterungen sind in Abb. 6.2 zu sehen.

#### **Druckversuch von einzelnen Messtaxeln (siehe Abb. 6.2 DV1)**

Der Halter (7) wird in die Halterung der Linearschiene eingesetzt und über eine Madenschraube fixiert. Anschließend kann das Miniaturspannfutter (4) auf den Halter (7) geschraubt und der Prüfstempel (5) im Spannfutter befestigt werden. Der Prüfling wird unter dem Prüfstempel ausgerichtet.

#### **Druckversuch zweier Sensorbahnen (siehe Abb. 6.2 DV2)**

Der Halter (7) wird in die Halterung der Linearschiene eingesetzt und über eine Madenschraube fixiert. Die zu prüfenden Sensorbahnen befinden sich auf kontaktierten Trägerplatten. Diese werden auf die Klebestreifen der Halter (1) und (2) geklebt. Halter (2) wird an Halter (7) angeschraubt und Halter (1) wird, mit den Leiterbahnen im rechten Winkel, darunter auf der Feinwaage platziert.

#### **Zugversuch von einzelnen Sensorbahnen (siehe Abb. 6.2 ZV)**

Der Klemmhalter (6) wird in die Halterung der Linearschiene eingesetzt und über eine Madenschraube angezogen. Das Miniaturspannfutter (4) wird auf den Halter (3) geschraubt und mittig auf der Feinwaage platziert. Die zu prüfende Sensorbahn wird unten mit dem Schnellspannfutter (4) und oben über die Klemmhalterung (6) festgeklemmt. Zuletzt sind die Widerstandsmessklemmen wie in Abb. 6.2 (ZV) zu sehen anzubringen.

## 7 Diskussion

Die Anforderungen an den Prüfstand aus Kap. 1.3 wurden wie folgt eingehalten:

### Sensorgrößen

Die maximale Sensorgröße ist anhand der Kuka Leichtbauroboter-Struktur, die mit taktilen Sensoren bestückt werden soll, vorgegeben. Sie hat die Abmessungen von  $25\text{cm} \cdot 15\text{cm} \cdot 12\text{cm}$  (siehe Abb. G.1).

Die Lineareinheiten der X und Y-Achse (KUT2005L-820-200-D3 [17, S. 461ff]) sind auf einen Verfahrweg von maximal 610mm und die Lineareinheit der Z-Achse (LX3005P-S2-T3056.4-600-OPS2 [17, S. 401ff]) auf 467mm ausgelegt. Der größere Verfahrweg wurde als zusätzlicher Bewegungsfreiraum für einen geplanten Zweiachsen-Rotationsprüfkopf, ähnlich der Rotationsachsen siehe Abb. 3.4, vorgesehen.

### Taxelanzahl 64

Die Verwendung der zwei VHF-Multiplexern HP 44472A in den Einschüben des modularen Schalt-/Steuerungs-Systems HP 3488A ermöglicht es, die 64 Sensormesspunkte über multiplexen der acht Sensorspalten und Sensorzeilen auf den Anschluss des Digitalmultimeters Fluke 45 zu führen, siehe Kap. 3.2.2 und Kap. 3.2.3. Die elektrische Kontaktierung des Sensorelementes zur Auswertung aller 64 Messpunkte ist somit möglich.

Die selbstständige mechanische Positionierung des Sensorelements unter dem Prüfstempel mithilfe des geplanten Linearachsportals konnte aufgrund zu langer Lieferzeiten im Bearbeitungszeitraum dieser Arbeit nicht realisiert werden.

### Wegauflösungsvermögen der statischen Anregung im 1/100mm Bereich

Die Wegauflösung in Prüfrichtung wird durch die Wegauflösung der Z-Achse bestimmt. Diese Achse wird mit einem Schrittmotor mit  $1,8^\circ$  mechanischer Winkelauflösung angetrieben, wobei die Schrittmotorkarte mikroschrittfähig ist und daher eine Winkelauflösung von  $0,11$  ermöglicht. Daraus resultiert eine Wegauflösung von  $0,002\text{mm}$ .

Mikroschrittbetrieb:  $1/16$  Schritt

Spindelsteigung LX3005P:  $5\text{mm}$

$$\Rightarrow \frac{1}{16} \cdot 1,8^\circ = 0,11^\circ$$

$$\Rightarrow 0,11^\circ \div 360^\circ \cdot 5\text{mm} = 0,002\text{mm}$$

(Z-Achse LX3005P-S2-T3056.4-600-OPS2 siehe [17, S. 401ff])

(Schrittmotor QMOT 6018-86-28-310 siehe [26])

### Schutz gegenüber Störquellen (z.B. Elektromagnetische Störfelder)

Eine Schirmungslösung der Messumgebung gegen elektromagnetische Störfelder wurde in Kap. 3.2.3 erarbeitet. Die technische Umsetzung konnte aufgrund von Lieferschwierigkeiten der Blechbiegekonstruktion im zeitlichen Rahmen der Bachelorarbeit nicht abgeschlossen werden.



### **Aufzeichnung der Messbedingungen (z.B. Temperaturverlauf)**

Wie in Abb. 3.12(b) zu sehen ist, wurde ein Messfühler des Präzisionsthermometers DOST-MANN P600 (siehe Tab. 3.1 in der Nähe des Prüflings angebracht. Die Temperaturerfassung (siehe Abb. 4.8(c)) findet mit LabVIEW® statt.

### **Orientierung der Sensorflächen im Raum (Sensor auf Roboterstruktur)**

Um in Zukunft die Prüfung von Sensorelementen auf ihrem Zielsystem, wie der Oberfläche einer Kuka LBR Struktur (siehe Abb. G.1), zu erlauben, wurden die Lineareinheiten und die Konstruktion dermaßen ausgelegt, dass ausreichend Platz und Verfahrensweg für die zusätzliche Montage von zwei Rotationsachsen (siehe Abb. 3.4) vorhanden ist. Diese Erweiterung in Kombination mit einer Kraftmessdose ermöglicht die Prüfung von Sensorelementen auf dreidimensionalen Freiformflächen, siehe Kap. 3.1.2.

## 8 Ausblick

Damit der Prüfstand auch als alleinstehendes System ohne Personal Computer eingesetzt werden kann, wurden Kommunikationsschnittstellen zu einer aktuell am Institut entwickelten Auswerteelektronik für taktile Sensoren (siehe [23]) vorgesehen. Dadurch wäre die Realisierung einer autonomen Sensorkalibrier- und Prüfeinheit denkbar, welche zuerst alle Kraftkurven des jeweiligen taktilen Sensorelements aufzeichnet und diese als Kalibrierwerte der Auswerteelektronik zur Verfügung stellt.

Des Weiteren wird das geplante Linearachsportal nicht nur für statische Sensortests Verwendung finden, sondern mit einer Erweiterung der Z-Achse um einen geeigneten Aktor für hohe Frequenzanregungen auch als dynamischer Teststand dienen.

Aufgrund von Lieferschwierigkeiten konnte, wie in den vorrausgehenden Kapiteln beschrieben, nicht das komplette geplante System im Zeitraum der Bachelorarbeit realisiert werden. Die Fertigstellung des vollständigen Prüfstandes wird jedoch als Werkstudententätigkeit am Deutschen Zentrum für Luft- und Raumfahrt fortgesetzt.

# Literaturverzeichnis

- [1] *Bussysteme für die Messtechnik*. In: BÖGE, W. und W. PLASSMANN (Hrsg.): *Vieweg Handbuch Elektrotechnik*, S. 797–802. Vieweg+Teubner, 2007. 10.1007/978-3-8348-9217-1\_65.
- [2] ALLEGRO MICROSYSTEMS, INC., 115 Northeast Cutoff, Box 15036 / Worcester, Massachusetts 01615-0036 (508) 853-5000: *A4983 DMOS Microstepping Driver with Translator*, 2007-2008. [http://www.allegromicro.com/Products/Motor-Driver-And-Interface-ICs/Bipolar-Stepper-Motor-Drivers/\\_media/Files/Datasheets/A4983-Datasheet.ashx](http://www.allegromicro.com/Products/Motor-Driver-And-Interface-ICs/Bipolar-Stepper-Motor-Drivers/_media/Files/Datasheets/A4983-Datasheet.ashx).
- [3] ALLEGRO MICROSYSTEMS, INC., 115 Northeast Cutoff, Box 15036 / Worcester, Massachusetts 01615-0036 (508) 853-5000: *Bipolar Stepper Motor Drivers*, 05 2012. <http://www.allegromicro.com/en/Products/Motor-Driver-And-Interface-ICs/Bipolar-Stepper-Motor-Drivers.aspx>.
- [4] ATMEL CORPORATION, 2325 Orchard Parkway / San Jose, CA 95131 USA: *AVR STK500 User Guide*, 2003. <http://www.atmel.com/Images/doc1925.pdf>.
- [5] BRUCE, M. F. und D. C. SINCLAIR: *The relationship between tactile thresholds and histology in the human finger*. Journal of Neurology, Neurosurgery, and Psychiatry, 43:235–242, 1980.
- [6] DAHIYA, R. S. und M. VALLE: *Tactile Sensing for Robotic Applications*. Sensors Peterborough NH, December, 2008.
- [7] DOEGE, E. und B.-A. BEHRENS: *Handbuch Umformtechnik: Grundlagen, Technologien, Maschinen (VDI-Buch) (German Edition)*. Springer, 2007.
- [8] EGEKVIST, H., P. BJERRING und L. ARENDT-NIELSEN: *Pain and mechanical injury of human skin following needle insertions*. European Journal of Pain, 3:41–49, 1999.
- [9] EISELE PNEUMATICS GMBH + CO. KG, Hermann-Hess-Straße 14-16 71332 Waiblingen Germany: *Eisele Kunststoffschläuche Katalogprogramm*.
- [10] EMIS GMBH, Zur Drehscheibe 4 92637 Weiden Germany: *SMC-TR-1000-Handbuch-v1*. <http://www.emisgmbh.de/antriebs/smc/smc-tr/SMC-TR-1000-Handbuch-v1.pdf>.
- [11] FLUKE DEUTSCHLAND GMBH, In den Engematten 14, 79286 Glottertal: *FLUKE 45 Multimeter mit Doppelanzeige Gebrauchsanleitung*, 2 Aufl., 04 1989.
- [12] HUG® INDUSTRIE-TECHNIK UND ARBEITSSICHERHEIT GMBH, Am Industrie-  
gleis 7, D-84030 Ergolding: *Internationale Vergleichstabelle für Baustähle*, 06 2012. <http://www.tabelle.info/baustahl.html>.
- [13] JOHN M. VRANISH, CROFTON; PRADEEP K. YADAV, B. B. O. M.: *MAGNETO-INDUCTIVE SENSOR FOR PERFORMING TACTILE AND PROXIMITY SENSING*. Techn. Ber., United States Patent, 08 1990. Patent Number: 4,950,987.
- [14] LUNDSTRÖM, R.: *Local vibrations - mechanical impedance of the human hand's glabrous skin*. Pergamon Press Ltd., 17:137–144, 1987.

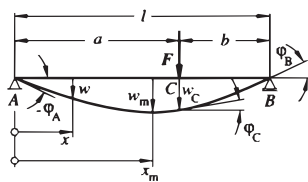
- [15] MAYR, M.: *Technische Mechanik: Anhang Tabellen Diagramme Formeln*. In: *Technische Mechanik: Statik-Kinematik-Kinetik-Schwingungen-Festigkeitslehre*. Hanser Verlag, 2008.
- [16] MÄDLER GMBH, Tränkestr. 6-8, 70597 Stuttgart: *Mädler-Katalog 39*, 3 Aufl., 01 2012.
- [17] MISUMI EUROPA GMBH, Katharina-Paulus-Str. 6 65824 Schwalbach am Taunus Germany: *MiSUMi Band 1/2 Mechanische Komponenten für Sondermaschinenbau & Montageautomation*, 2011-2012.
- [18] MISUMI EUROPA GMBH, Katharina-Paulus-Str. 6 65824 Schwalbach am Taunus Germany: *MiSUMi Band 2/2 Mechanische Komponenten für Sondermaschinenbau & Montageautomation*, 2011-2012.
- [19] PFEIFER, C.: *Entwurf und Konstruktion eines Kleinstmengenmischgeräts für Silikonelastomere*. Diplomarbeit, Hochschule für angewandte Wissenschaften Augsburg, Deutsches Zentrum für Luft- und Raumfahrt, 2010.
- [20] REIF, K.: *Bussysteme*. In: REIF, K. (Hrsg.): *Batterien, Bordnetze und Vernetzung*, S. 134–207. Vieweg+Teubner, 2010. 10.1007/978-3-8348-9713-8\_7.
- [21] SACK, L.: *Elektrische Antriebstechnik*. In: GEVATTER, H.-J. und U. GRÜNHaupt (Hrsg.): *Handbuch der Mess- und Automatisierungstechnik in der Produktion*, VDI-Buch, S. 657–692. Springer Berlin Heidelberg, 2006. 10.1007/3-540-34823-9\_22.
- [22] SANYO DENKI GERMANY GMBH, Frankfurter Str.92, D-65760 Eschborn, Germany: *SANMOTION 2-PHASE STEPPING SYSTEMS F2*, 3 Aufl., 06 2012. CATALOG No. S0832B010 '12.1.IT.
- [23] SCHNEIDER, D.: *Unveröffentlichtes Dokument, Erweiterung einer Auswerteelektronik für taktile Sensoren zum multimodalen Sensorsystem*. Bachelorarbeit, Hochschule für angewandte Wissenschaften Augsburg, Deutsches Zentrum für Luft- und Raumfahrt, 2012.
- [24] SCHNÖS, F. Bachelor thesis, Technische Universität München, Munich, 2011.
- [25] STROHMAYR, M.: *Artificial Skin in Robotics*. Doktorarbeit, Karlsruhe Institute of Technology (KIT), 2012. submitted May 03, 2012.
- [26] TRINAMIC MOTION CONTROL GMBH & CO. KG, Waterloohein 5, 22769 Hamburg, Germany: *QMOT QSH6018 MANUAL*, 1.07 Aufl., 02 2012.
- [27] WETTELS, N., V. J. SANTOS, R. S. JOHANSSON und G. E. LOEB: *Biomimetic Tactile Sensor Array*. *Advanced Robotics*, 22(8):829–849, 2008.
- [28] WOLFSPERGER, H.: *Elektromagnetische Schirmung: Theorie und Praxisbeispiele*. Springer Verlag, 2008.
- [29] WWW.MIKROCONTROLLER.NET, Andreas Schwarz, Drausnickstr. 1a, 91052 Erlangen: *AVR-GCC-Tutorial*, 01 2012.
- [30] ZWICK GMBH & CO. KG, August-Nagel-Str. 11, D-89079 Ulm: *Zwick Materialprüfung Tisch-Prüfmaschinen Allround-Line*, 05 2012.

# Anhang

## A Mechanik Formelsammlung

### A 14.1 Biegeverformungen von statisch bestimmten Trägern mit konstantem Querschnitt

Vereinbarungen:



$\xi = \frac{x}{l}$ ;  $\alpha = \frac{a}{l}$ ;  $\beta = \frac{b}{l}$ : Bezogene Längen (zur kompakten Schreibweise der Biegelinie)

$w$ : Durchbiegung an der Stelle  $x$  bzw.  $x_1$  oder  $x_2$  (nach unten positiv)

$w_C$ : Durchbiegung am Kraft- oder Momentenangriffspunkt bzw. am freien Ende (nach unten positiv)

$w_m$ : maximale Durchbiegung zwischen zwei Lagern (die Richtung ergibt sich aus der jeweiligen Fallskizze)

$\phi_A, \phi_B, \phi_C$ : Biegewinkel an den Stellen  $A, B$  und  $C$  (im Gegenuhrzeigersinn positiv)

Nr.	Biegefall	Biegelinie	Durchbiegung	Biegewinkel
1		$w = \frac{Fl^3}{6EI} (2 - 3\xi + \xi^3)$	$w_A = \frac{Fl^3}{3EI}$	$\phi_A = \frac{Fl^2}{2EI}$
2		$w = \frac{Ml^2}{2EI} (1 - 2\xi + \xi^2)$	$w_A = \frac{Ml^2}{2EI}$	$\phi_A = \frac{Ml}{EI}$
3		$w = \frac{ql^4}{24EI} (3 - 4\xi + \xi^4)$	$w_A = \frac{ql^4}{8EI}$	$\phi_A = \frac{ql^3}{6EI}$
4		$w = \frac{ql^4}{120EI} (4 - 5\xi + \xi^5)$	$w_A = \frac{ql^4}{30EI}$	$\phi_A = \frac{ql^3}{24EI}$

Abbildung A.1: Biegefälle Tabelle Teil 1 [15, A14.1]

A 14.1 Biegeverformungen von statisch bestimmten Trägern mit konstantem Querschnitt (Fortsetzung)

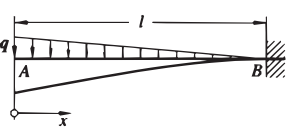
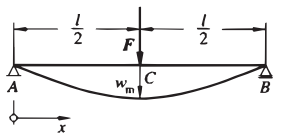
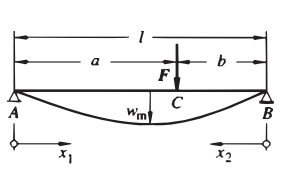
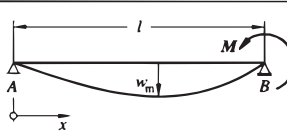
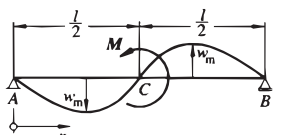
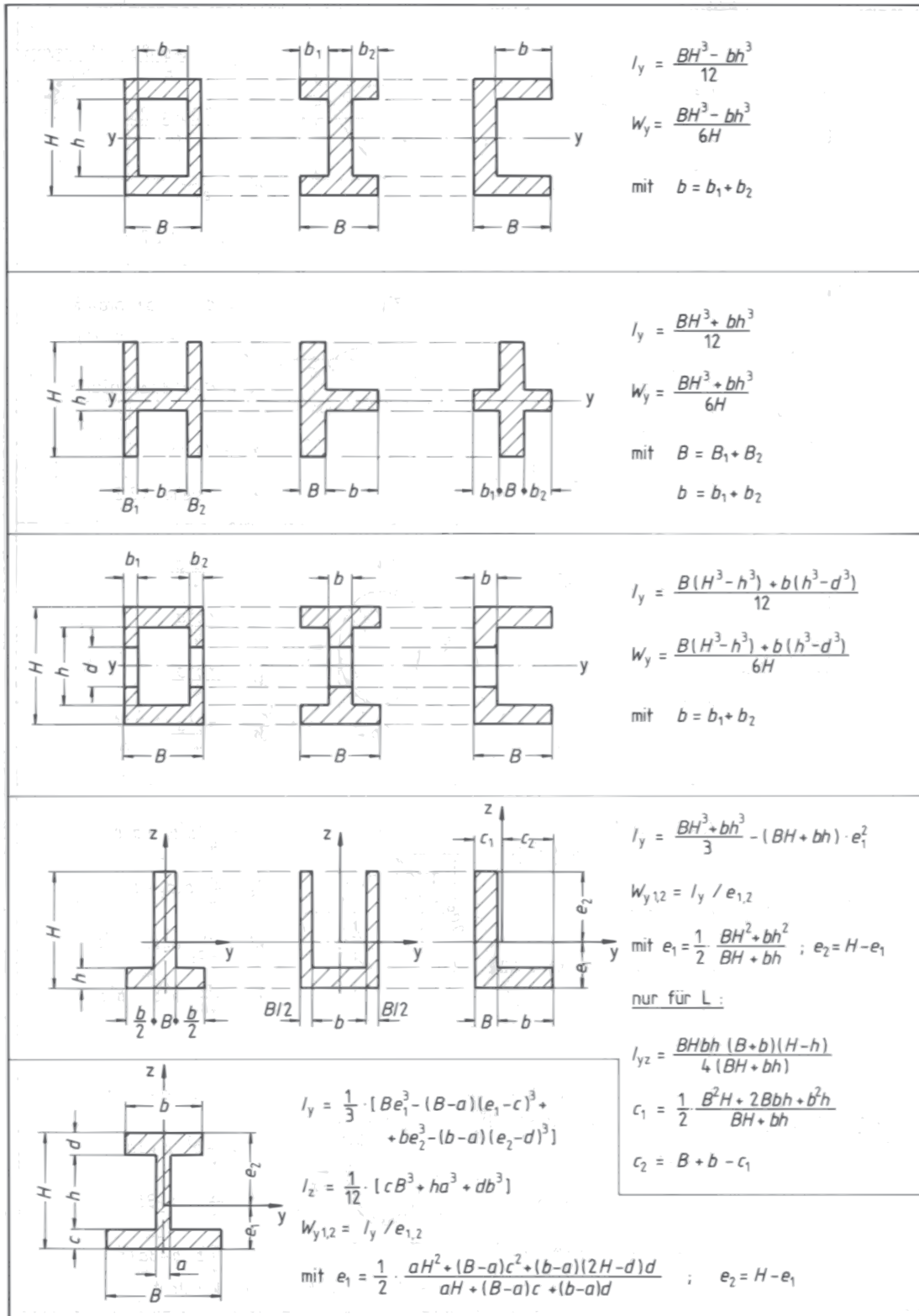
Nr.	Biegefall	Biegelinie	Durchbiegung	Biegewinkel
5		$w = \frac{ql^4}{120EI} (11 - 15\xi + 5\xi^4 - \xi^5)$	$w_A = \frac{11}{120} \frac{ql^4}{EI}$	$\varphi_A = \frac{ql^3}{8EI}$
6		$0 \leq \xi \leq 1/2:$ $w = \frac{Fl^3}{48EI} \xi (3 - 4\xi^2)$	$w_C = w_m = \frac{Fl^3}{48EI}$	$-\varphi_A = \varphi_B = \frac{Fl^2}{16EI}$
7		$0 \leq \xi_1 \leq \alpha:$ $w = \frac{Fl^3\beta}{6EI} \xi_1 (1 - \beta^2 - \xi_1^2)$ $0 \leq \xi_2 \leq \beta:$ $w = \frac{Fl^3\alpha}{6EI} \xi_2 (1 - \alpha^2 - \xi_2^2)$	$w_C = \frac{Fa^2b^2}{3EI}$ für $a \geq b$ : $w_m = \frac{Fb\sqrt{(l^2 - b^2)^3}}{9\sqrt{3}EI}$ in $x_m = \sqrt{\frac{1}{3}(l^2 - b^2)}$	$\varphi_A = -\frac{Fab}{6EI}(l + b)$ $\varphi_B = \frac{Fab}{6EI}(l + a)$ $\varphi_C = \frac{Fab}{3EI}(a - b)$
8		$w = \frac{Ml^2}{6EI} \xi (1 - \xi^2)$	$w_m = \frac{Ml^2}{9\sqrt{3}EI}$ in $x_m = \frac{l}{\sqrt{3}}$	$\varphi_A = -\frac{Ml}{6EI}$ $\varphi_B = \frac{Ml}{3EI}$
9		$0 \leq \xi \leq 1/2:$ $w = \frac{Ml^2}{24EI} \xi (1 - 4\xi^2)$	$w_m = \frac{\sqrt{3}}{216} \frac{Ml^2}{EI}$ in $x_m = \frac{l}{2\sqrt{3}}$	$\varphi_A = \varphi_B = -\frac{Ml}{24EI}$ $\varphi_C = \frac{Ml}{12EI}$

Abbildung A.2: Biegefälle Tabelle Teil 2 [15, A14.1]

A7.1 Fortsetzung 2



A<sup>7.1</sup>

Abbildung A.3: Flächenmomente 2. Grades [15, A7.1]

A9.1 Elastizitätsmodul E

WERKSTOFF	E kN/mm <sup>2</sup>	WERKSTOFF	E kN/mm <sup>2</sup>
Diamant	1000	Aluminiumlegierungen	60-80
Osmium	560	Porzellan	60-80
Wolframcarbid	450-650	Glas, allgemein	39-98
Iridium	528	Quarzglas	62-75
Hartmetall	343-667	Kalkstein	40-90
Kohlenstofffaser (hochmodul)	380-550	Granit	62
Cermets	400-530	Antimon	56
Siliziumcarbid	450	Zinn	55
Borfaser	420-450	Sinterbronze C 50	50
Wolfram	407	Magnesium	44
Aluminiumoxid-Keramik	350-420	Magnesiumlegierungen	40-45
Aluminiumoxid	370	Fiberglas	35-45
Dispersions-Keramik	360	Beton B25	30
Molybdän	334	Aluminiumtitanat-Keramik	30
Titancarbid	250-380	Flachs	26
Magnesiumoxid	250	Graphit	5-30
Chrom	250	Blei	16
Zirkonoxid	220	Buche faserparallel	14
Zirkonoxid-Keramik	210	Eiche faserparallel	13
Kobalt	210	Kiefer faserparallel	12
Nickel	210	Fichte faserparallel	11
Baustahl	210	Sperrholz	4-16
Nichtrostende ferrit. Stähle	206-212	Naturseide	10
Nichtrostende martensit. Stähle	206-212	Eis	10
Eisen	206	Melaminharz	5-9
Federstähle	206	Baumwolle	4,8
Mangan	200	Hartgummi SH 60	4
Nichtrostende austenit. Stähle	190-200	Polyimid (PJ)	3-5
Tantal	186	Polymethylmetacrylat (PMMA)	3-4
Nickellegierungen	130-234	Polystyrol (PS)	3-4
Platin	170	Araldit	3,2
Mullit	145	Polyamid (PA)	2-4
CFK	70-200	Nylon-Kunstseide.orientiert	2-4
Aramidfaser	130	Polyesterharz	1-5
Vanadium	130	Polyethylenterephthalat (PETP)	1-5
Kupfer	125	Plexiglas	2,7-3,2
Palladium	124	Acryl	1,6-3,4
Bor	121	Expoxidharz (EP)	2-3
Gusseisen	60-180	Polycarbonat (PC)	2-3
Bronze	105-124	Buche radial	2,3
Titan	110	Polyvinylchlorid (PVC)	1-3
Niob	105	Eiche radial	1,6
Messing	78-123	Polyethylen hoher Dichte (PE-HD)	1,2
Silizium	100	Kiefer radial	1
Zink	70-130	Fichte radial	0,8
Steatit	88-98	Polypropylen (PP)	0,4-0,9
Silber	80	Polytetrafluorethylen (PTFE)	0,4
Gold	78	Phenolharz	0,3
Marmor	60-90	Polyethylen kleiner Dichte (PE-LD)	0,2
Sicherheitsglas	62-86	Silikonkautschuk	0,01-0,1
Aluminium	71	Schaumstoff	0,001-0,1
Glasfaser	70		

A<sup>9.1</sup>

Abbildung A.4: Elastizitätsmodul E [15, A14.1]



B EMF Schirmungsblech technische Zeichnungen

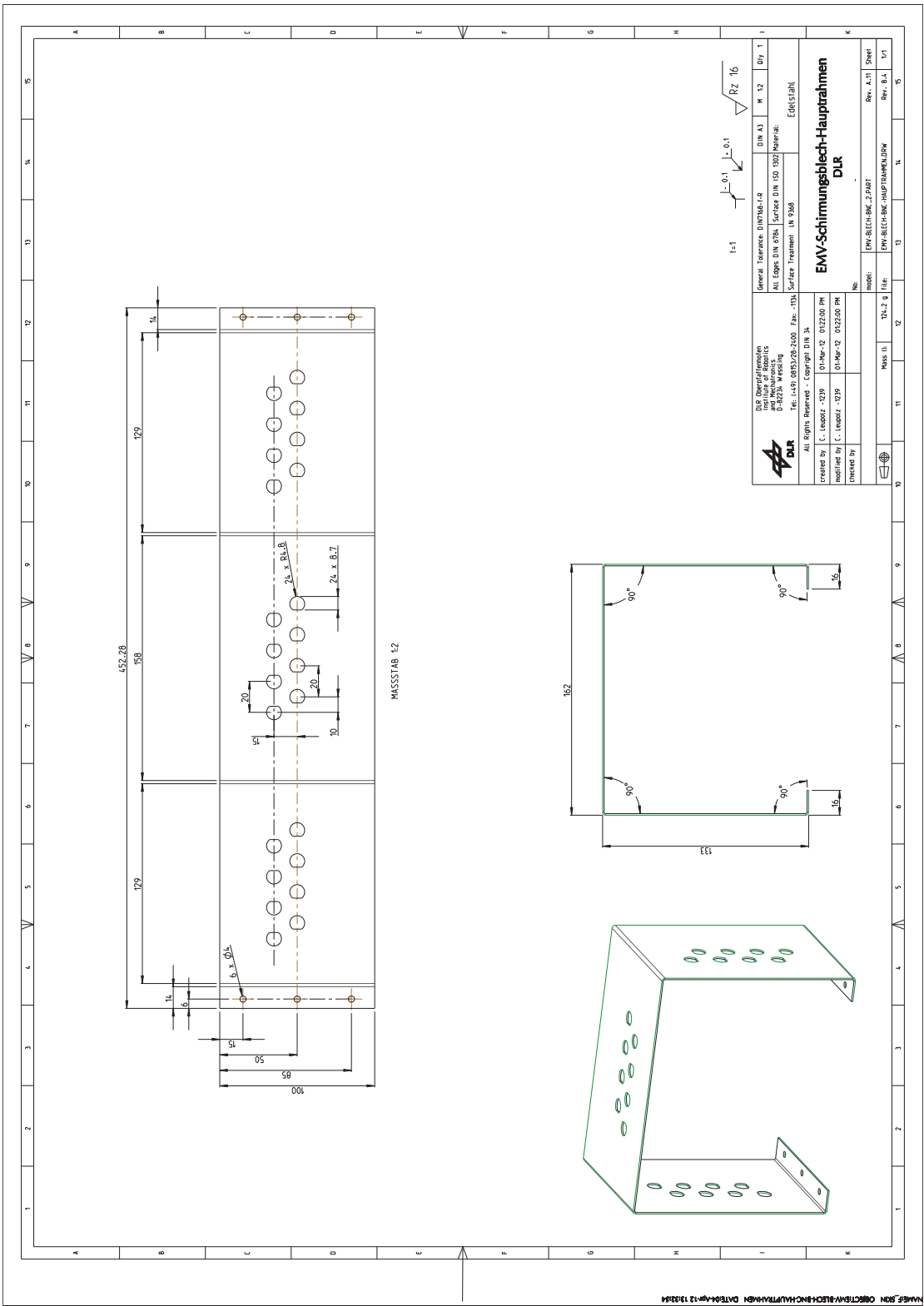


Abbildung B.1: EMF Schirmungsblech Hauptrahmen





## C Quellcode Steuerungssoftware “Stepper Control”

Nachfolgend finden sich die wesentlichen Abschnitte des in C verwirklichten Steuerungsprogramm mit ausführlicher Kommentierung. (Vollständiger Quellcode siehe Quellcode C.2)

Quellcode C.1: Ausführliche Dokumentation Steuerungssoftware “Stepper Control”

```

1  //...//      <-- Markiert ausgelassene Programmabschnitte, die nicht zum
2              // Verständnis des Hauptprogramms vonnöten sind.
3              // (Vollständiger Sourcecode siehe Anhang)
4
5  #define CYCLES_PER_S 31250          // 16000000/256/2
6                                      // (clk)/(prescaler)/(2 Pegelwechsel = 1 Step)
7  //...//
8
9  // #define perf_log      // Performanceausgabe: Gibt einen der Zykluszeit direkt
10 #ifdef perf_log          //              proportionalen Wert aus. Sie dient somit der
11 uint8_t perf_loop=0;     //              Performanceüberprüfung während der Entwicklung
12 uint32_t perf_count=0;
13 #endif
14
15 // Die Variablen tcnt1_x/y/z speichern den Zeitpunkt, zudem die jeweilige Achse
16 // den letzten Schritt ausführte, um damit die Zeitdifferenz bis zum nächsten
17 // Schritt bestimmen zu können. (siehe Codezeile 119)
18 // Die Variablen offset_x/y/z werden für eine Zähleroffsetkorrektur nach einem
19 // Zählerüberlauf benötigt. (siehe Codezeile 168 ff.)
20 volatile uint16_t tcnt1_x, tcnt1_y, tcnt1_z, offset_x = 0, offset_y = 0, offset_z = 0;
21
22
23 /* INITIALIZE */
24 void port_init(){//...//}          // Voreinstellung der I/O-Register
25 void timer_init(){//...//}        // Voreinstellung der Zählerregister
26 void uart_init(){//...//}         // Voreinstellung der Uartschnittstelle
27
28 void uart_putc(unsigned char uart_char){//...//}
29 void uart_puts(char *uart_string){//...//}
30
31
32 int main()          // --- BEGINN DES HAUPTPROGRAMMS ---
33 {
34     port_init();
35     timer_init();
36     uart_init();
37
38     sei();           // Interrupts aktivieren
39
40     OCR1A = 65535;   // Wert ab dem der Zähler zurückgesetzt wird
41
42     _delay_ms(10);
43
44     // Ausgabe über Uart: Versionsnummer & Datum
45     uart_puts("UART Stepper Control V1.1 (05.02.2012)\n");
46     _delay_ms(10);
47
48     // Ausgabe über Uart: Bedienhinweise
49     uart_puts("Input: stepper (x,y,z) | steps (3200 steps/round)");
50     uart_puts(" | dir (0=ccw 1=cw) | speed (steps/s)\n");
51
52     struct stepper_motor { // Struct eines einzelnen Schrittmotors
53         uint32_t steps;
54         uint8_t dir;
55         uint16_t speed;
56     };

```

```

56 struct axis { // Struct für ein Drei-Achs-System
57     struct stepper_motor x_axis;
58     struct stepper_motor y_axis;
59     struct stepper_motor z_axis;
60 } coordinate_table = {{0,1,1},{0,1,1},{0,1,1}}; // Default Werte
61
62 while (1) // --- HAUPTSCHLEIFE DES MIKROCONTROLLERS ---
63 {
64     if(uart_str_complete) // Aufruf wenn neue Benutzereingabe vorhanden
65     {
66         struct stepper_motor* stepper_motor_pt = 0; // Zeiger vom Typ Schrittmotor
67 // wird nachfolgend einer
68 // Schrittmotorachse zugeordnet
69
70 // Zuordnung Schrittmotor-Achse
71 switch(uart_string[0]) // uart_string[0] entspricht dem
72 { // ersten Zeichen der Benutzereingabe
73     case 'x': stepper_motor_pt = &coordinate_table.x_axis; break;
74     case 'y': stepper_motor_pt = &coordinate_table.y_axis; break;
75     case 'z': stepper_motor_pt = &coordinate_table.z_axis; break;
76 }
77
78 // Ordnet die restlichen Benutzereingaben, zu fahrende Schritte, Richtung
79 // und Geschwindigkeit, den entsprechenden Variablen der Motorachsen zu.
80 sscanf((char*)uart_string, "%s %lu %hhu %u", &stepper_motor_pt->steps,
81 // &stepper_motor_pt->dir, &stepper_motor_pt->speed);
82
83 stepper_motor_pt->steps *= 2; // 2 Pegelwechsel = 1 Step
84 // Schrittzahl wird verdoppelt, da ein
85 // Schritt aus einem high Pegel und einem
86 // low Pegel besteht
87
88 stepper_motor_pt->speed = CYCLES_PER_S/stepper_motor_pt->speed;
89 // CYCLES_PER_S steht für die Anzahl der Inkrementierungen des
90 // Zählers Timer1 pro Sekunde.
91 // In stepper_motor_pt->speed befindet sich die Benutzereingabe
92 // der Geschwindigkeit in Schritte pro Sekunde.
93
94 // Benötigt wird jedoch die Wartezeit zwischen zwei Motorschritten
95 // als Anzahl der Inkrementierungen des Zählers Timer1.
96 // Folglich müssen die Anzahl der Timer1 Inkrementierungen pro Sekunde
97 // durch die gewünschten Schritte pro Sekunde dividiert werden.
98
99 //...//
100 }
101
102 // --- ANSTEUERUNG DER X-ACHSE ---
103 // Wird ausgeführt, wenn delta Zählerstand
104 // (TCNT1 (aktueller Zählerstand) - tcnt1_x (alter Zählerstand))
105 // größer als die Wartezeit zwischen zwei Motorschritten ist.
106 if((((TCNT1 - tcnt1_x) + offset_x) > coordinate_table.x_axis.speed)
107 // && coordinate_table.x_axis.steps)
108 {
109     if(coordinate_table.x_axis.dir) // Schaltet den Richtungspin entsprechend
110     { // der Benutzereingabe
111         PORTA &=(1<<PA0); // Direction cw
112     }
113     else
114     {
115         PORTA |= (1<<PA0); // Direction ccw
116     }
117     PORTA ^= (1<<PA1); // Step PIN togglen
118     coordinate_table.x_axis.steps--; // Schrittezähler um eins verringern
119     tcnt1_x = TCNT1; // aktuelle Zeit sichern um naechstes dt bilden zu können
120     offset_x = 0; // Offset, falls vorhanden von Timerüberlauf,

```

```

121     }                                     // nun wieder zurücksetzen, da tcnt1_x nicht mehr
122     // "ueber das Register übersteht"
123
124
125     // --- ANSTEUERUNG DER Y-ACHSE --- (vergleiche X-ACHSE)
126     if((((TCNT1 - tcnt1_y) + offset_y) > coordinate_table.y_axis.speed)
127         && coordinate_table.y_axis.steps)
128     {
129         if(coordinate_table.y_axis.dir){PORTA ^= (1<<PA2);}
130         else {PORTA |= (1<<PA2);}
131         PORTA ^= (1<<PA3);
132         coordinate_table.y_axis.steps--;
133         tcnt1_y = TCNT1;
134         offset_y = 0;
135     }
136
137     // --- ANSTEUERUNG DER Z-ACHSE --- (vergleiche X-ACHSE)
138     if((((TCNT1 - tcnt1_z) + offset_z) > coordinate_table.z_axis.speed)
139         && coordinate_table.z_axis.steps)
140     {
141         if(coordinate_table.z_axis.dir){PORTA ^= (1<<PA4);}
142         else {PORTA |= (1<<PA4);}
143         PORTA ^= (1<<PA5);
144         coordinate_table.z_axis.steps--;
145         tcnt1_z = TCNT1;
146         offset_z = 0;
147     }
148
149
150     #ifdef perf_log                       // siehe oben Performanceausgabe
151     if(perf_loop == 1)
152     {
153         char string_temp[20];
154         ultoa(perf_count, string_temp, 10);
155         uart_puts(string_temp);
156         uart_puts("\n");
157         perf_loop = 2;
158         perf_count = 0;
159     }
160     else if(!perf_loop){perf_count++;}
161     #endif
162 }
163 }
164
165
166 ISR(TIMER1_COMPA_vect) // Interrupt-Service-Routine Zähler Timer1 überlauf
167 {
168     if(tcnt1_x){offset_x = OCR1A - tcnt1_x;} // Falls ein Zählerstand gespeichert
169     if(tcnt1_y){offset_y = OCR1A - tcnt1_y;} // wurde, muss ein Offsetwert berechnet
170     if(tcnt1_z){offset_z = OCR1A - tcnt1_z;} // werden.
171                                           // OCR1A entspricht dem maximalen
172                                           // Zählerstand des Timer1 Registers
173     tcnt1_x = 0; // Zählerstandvariablen rücksetzen
174     tcnt1_y = 0;
175     tcnt1_z = 0;
176
177
178     #ifdef perf_log                       // siehe oben Performanceausgabe
179     perf_loop = !perf_loop;
180     #endif
181 }
182
183
184 ISR(USART_RXC_vect){//...//} // Interrupt-Service-Routine Uart Empfangspuffer

```

## Quellcode C.2: Steuerungssoftware "Stepper Control" vollständig

```

1  #include <avr/io.h>
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <string.h>
5  #include <util/delay.h>
6  #include <avr/interrupt.h>
7
8  #define BAUD 9600UL
9  #include <util/setbaud.h>
10
11
12 #define UBRR_VAL ((F_CPU+BAUD*8)/(BAUD*16)-1) // clever runden
13 #define BAUD_REAL (F_CPU/(16*(UBRR_VAL+1))) // Reale Baudrate
14 #define BAUD_ERROR ((BAUD_REAL*1000)/BAUD) // Fehler in Promille, 1000 = kein Fehler.
15
16 #if ((BAUD_ERROR<990) || (BAUD_ERROR>1010))
17     #error Systematischer Fehler der Baudrate grösser 1% und damit zu hoch!
18 #endif
19
20 #define UART_MAXSTRLEN 30
21 #define CYCLES_PER_S 31250 // 16000000/256/2
22 // (clk)/(prescaler)/(2 Pegelwechsel = 1 Step)
23
24 volatile uint8_t uart_str_complete = 0; // 1 .. String komplett empfangen
25 volatile uint8_t uart_str_count = 0;
26 volatile char uart_string[UART_MAXSTRLEN + 1] = "";
27
28 // #define perf_log // Performanceausgabe
29 #ifdef perf_log
30     uint8_t perf_loop=0;
31     uint32_t perf_count=0;
32 #endif
33
34
35 volatile uint16_t tcnt1_x, tcnt1_y, tcnt1_z, offset_x = 0, offset_y = 0, offset_z = 0;
36
37
38
39 /* INITIALIZE */
40 void port_init()
41 {
42     DDRA |= 0xff;
43     PORTA |= 0x00;
44     DDRB |= 0xff;
45     PORTB |= 0xff;
46     DDRD |= (1<<PD1);
47     DDRD &=~(1<<PD0);
48 }
49
50 void timer_init()
51 {
52     /*TIMER*/
53     TCCR1B |= (1<<WGM12); // CTC Mode4 (TOP = OCR1A)
54     TCCR1B |= (1<<CS12); // clk/256
55     TIMSK |= (1<<OCIE1A); // Interrupt Output Compare A Match
56 }
57
58 void uart_init()
59 {
60     UBRRH = UBRR_VAL >> 8;
61     UBRRL = UBRR_VAL & 0xFF;
62
63     UCSRB |= (1<<TXEN)|(1<<RXEN)|(1<<RXCIE); // UART TX & RX einschalten

```

```

64     UCSRC = (1<<URSEL)|(1<<UCSZ1)|(1<<UCSZ0);           // Asynchron 8N1
65 }
66
67 void uart_putc(unsigned char uart_char)
68 {
69     while (!(UCSRA & (1<<UDRE)))                          // warten bis Senden moeglich
70     {
71     }
72     UDR = uart_char;
73 }
74
75
76 void uart_puts(char *uart_string)
77 {
78     while (*uart_string) // so lange *uart_string != '\0' (ungleich String-Ende)
79     {
80         uart_putc(*uart_string);
81         uart_string++;
82     }
83 }
84
85
86 int main()
87 {
88     port_init();
89     timer_init();
90     uart_init();
91
92     sei();
93
94     OCR1A = 65535;
95
96     _delay_ms(10);
97     uart_puts("UART Stepper Control V1.1 (05.02.2012)\n");
98     _delay_ms(10);
99     uart_puts("Input: stepper (x,y,z) | steps (3200 steps/round) | dir (0=ccw 1=cw) |
100                speed (steps/s)\n");
101
102     struct stepper_motor {
103         uint32_t steps;
104         uint8_t dir;
105         uint16_t speed;
106     };
107
108     struct axis {
109         struct stepper_motor x_axis;
110         struct stepper_motor y_axis;
111         struct stepper_motor z_axis;
112     } coordinate_table = {{0,1,1},{0,1,1},{0,1,1}};
113
114     while (1)
115     {
116         if(uart_str_complete)
117         {
118             struct stepper_motor* stepper_motor_pt = 0;
119
120             switch(uart_string[0]) // Zuordnung Stepper-Achse
121             {
122                 case 'x': stepper_motor_pt = &coordinate_table.x_axis;
123                             break;
124                 case 'y': stepper_motor_pt = &coordinate_table.y_axis;
125                             break;
126                 case 'z': stepper_motor_pt = &coordinate_table.z_axis;
127                             break;

```



```

125     }
126     sscanf((char*)uart_string, "%*s %lu %hhu %u", &stepper_motor_pt->
127         steps, &stepper_motor_pt->dir, &stepper_motor_pt->speed);
128     stepper_motor_pt->steps *= 2; // 2 Pegelwechsel = 1 Step
129     stepper_motor_pt->speed = CYCLES_PER_S/stepper_motor_pt->speed;
130
131     uart_str_complete = 0;
132
133     char string_temp[20]; // DEBUG Delta t Step Toggle
134     utoa(stepper_motor_pt->speed, string_temp, 10);
135     uart_puts(string_temp);
136     uart_puts("\n");
137 }
138
139 if((((TCNT1 - tcnt1_x) + offset_x) > coordinate_table.x_axis.speed) &&
140     coordinate_table.x_axis.steps) // CYCLES_PER_S = 7812 = 1 Step/s;
141     stepper[0][0] <- noch Schritte ueber?
142 {
143     if(coordinate_table.x_axis.dir){PORTA &=~(1<<PA0);} // Direction cw
144     else {PORTA |= (1<<PA0);} // Direction ccw
145     PORTA ^= (1<<PA1); // Step PIN togglen
146     coordinate_table.x_axis.steps--; // Uebrige Steps um eins
147     verringern
148     tcnt1_x = TCNT1; // aktuelle Zeit sichern um naechstes
149     Delta t bilden zu koennen
150     offset_x = 0; // Offset, falls vorhanden von
151     Timerueberlauf,
152 } // nun wieder zuruecksetzen, da tcnt1_x
153 nicht mehr "ueber das Register geht"
154
155 if((((TCNT1 - tcnt1_y) + offset_y) > coordinate_table.y_axis.speed) &&
156     coordinate_table.y_axis.steps)
157 {
158     if(coordinate_table.y_axis.dir){PORTA &=~(1<<PA2);}
159     else {PORTA |= (1<<PA2);}
160     PORTA ^= (1<<PA3);
161     coordinate_table.y_axis.steps--;
162     tcnt1_y = TCNT1;
163     offset_y = 0;
164 }
165
166 if((((TCNT1 - tcnt1_z) + offset_z) > coordinate_table.z_axis.speed) &&
167     coordinate_table.z_axis.steps)
168 {
169     if(coordinate_table.z_axis.dir){PORTA &=~(1<<PA4);}
170     else {PORTA |= (1<<PA4);}
171     PORTA ^= (1<<PA5);
172     coordinate_table.z_axis.steps--;
173     tcnt1_z = TCNT1;
174     offset_z = 0;
175 }
176
177 #ifdef perf_log
178 if(perf_loop == 1)
179 {
180     char string_temp[20];
181     ultoa(perf_count, string_temp, 10);
182     uart_puts(string_temp);
183     uart_puts("\n");
184     perf_loop = 2;
185     perf_count = 0;
186 }
187 else if(!perf_loop)
188 {
189     perf_count++;
190 }

```

```

181         }
182         #endif
183
184     }
185 }
186
187
188 ISR(TIMER1_COMPA_vect)
189 {
190     if(tcnt1_x){offset_x = OCR1A - tcnt1_x;}
191     if(tcnt1_y){offset_y = OCR1A - tcnt1_y;}
192     if(tcnt1_z){offset_z = OCR1A - tcnt1_z;}
193     tcnt1_x = 0;
194     tcnt1_y = 0;
195     tcnt1_z = 0;
196
197     #ifdef perf_log
198     perf_loop = !perf_loop;
199     #endif
200 }
201
202
203 ISR(USART_RXC_vect)
204 {
205     unsigned char nextChar;
206
207     // Daten aus dem Puffer lesen
208     nextChar = UDR;
209     if(uart_str_complete == 0)        // wenn uart_string gerade in Verwendung, neues
        Zeichen verwerfen
210     {
211         // Daten werden erst in uart_string geschrieben, wenn nicht String-Ende/
        max Zeichenlänge
212         // erreicht ist/string gerade verarbeitet wird
213         if(nextChar != '\n' && nextChar != '\r' && uart_str_count < UART_MAXSTRLEN
        - 1)
214         {
215             uart_string[uart_str_count] = nextChar;
216             uart_str_count++;
217         }
218         else
219         {
220             uart_string[uart_str_count] = '\0';
221             uart_str_count = 0;
222             uart_str_complete = 1;
223         }
224     }
225 }

```

(USART Routine siehe [29])

## D Quellcode Matlabskript zur Messwertaufbereitung

### Quellcode D.1: Matlabskript "Auswertung Zugversuch"

```
1 %Einlesen der Messprotokolldatei "data.lvm".
2 %Daten in einer Zeile sind durch Tabulator getrennt ("\t").
3 %Die ersten 23 Zeilen sind Kopf des Messprotokolls.
4 %    -> Anfang der Messdaten in Zeile "24" Zeilenanfang "0".
5 %Erzeugen der Matrix x.
6 x = dlmread('data.lvm','\t',24,0)
7
8 %Umrechnung in Einheit Newton
9 %Multiplikation der zweiten Messspalte (Datenreihe Waage [g])
10 %mit [9,81 m/s^2 / 1000]
11 x(:,2) = x(:,2)*0.00981
12
13 %Daten der Spalte "2" und "3" plotten.
14 plot(x(10:end,2),x(10:end,3),'LineWidth',2)
15
16 %Achsbeschriftungen
17 xlabel('Force [N]','FontSize',16)
18 ylabel('Resistance [\Omega]','FontSize',16)
19
20 %(Optional) Legende hinzufügen
21 legend('force/resistance','NorthEast')
22 set(legend,'FontSize',16)
23
24 %Ablegen der Grafik in den drei
25 %Dateiformaten *.fig, *.png und *.pdf.
26 saveas(gcf,'Plot.fig')
27 saveas(gcf,'Plot.png')
28 saveas(gcf,'Plot.pdf')
```

Die Matlabskripte zur Auswertung von Druckversuchen sind analog dem Zugversuch aufgebaut.

## E LabVIEW® Blockschaltplan Funktionsgruppen

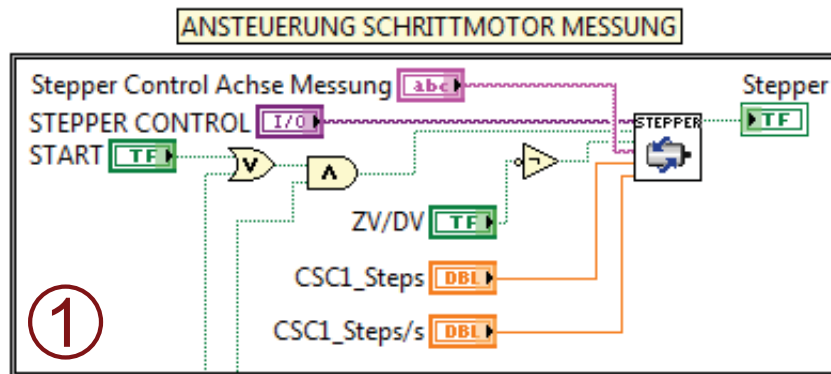


Abbildung E.1: LabVIEW® Blockschaltplan „Ansteuerung Schrittmotor Messung“

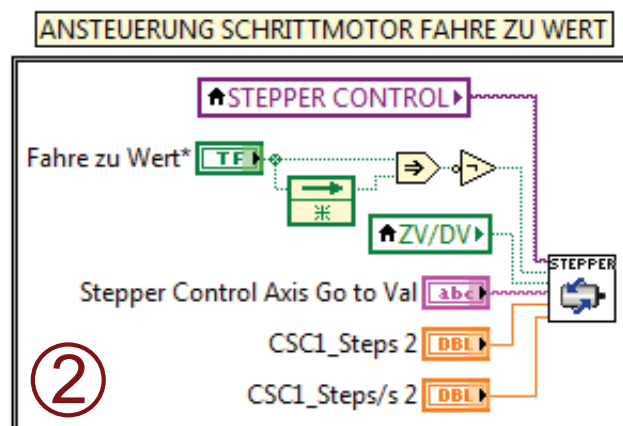


Abbildung E.2: LabVIEW® Blockschaltplan „Ansteuerung Schrittmotor fahre zu Wert“

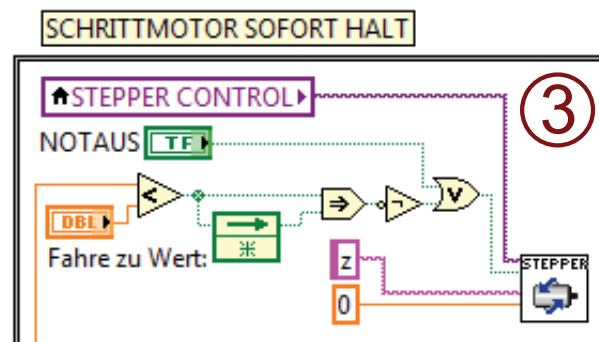


Abbildung E.3: LabVIEW® Blockschaltplan „Ansteuerung Schrittmotor fahre zu Wert“

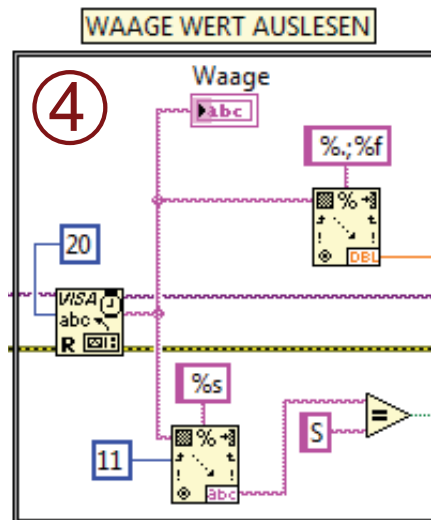


Abbildung E.4: LabVIEW® Blockschaltplan „Waage Wert auslesen“

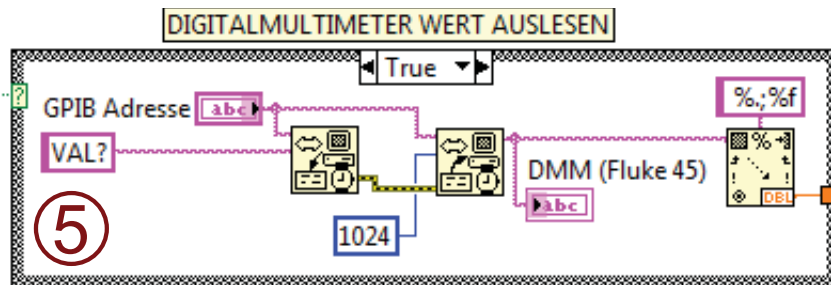


Abbildung E.5: LabVIEW® Blockschaltplan „Digitalmultimeter Wert auslesen“

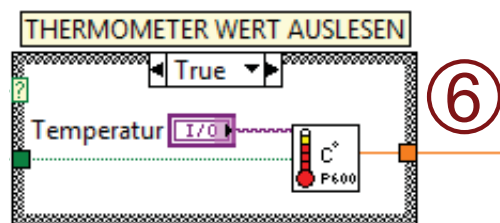


Abbildung E.6: LabVIEW® Blockschaltplan „Thermometer Wert auslesen“

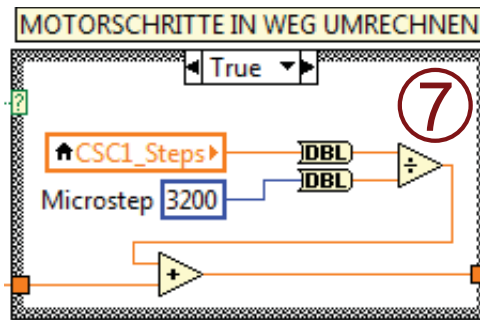


Abbildung E.7: LabVIEW® Blockschaltplan „Motorschritte in Weg umrechnen“

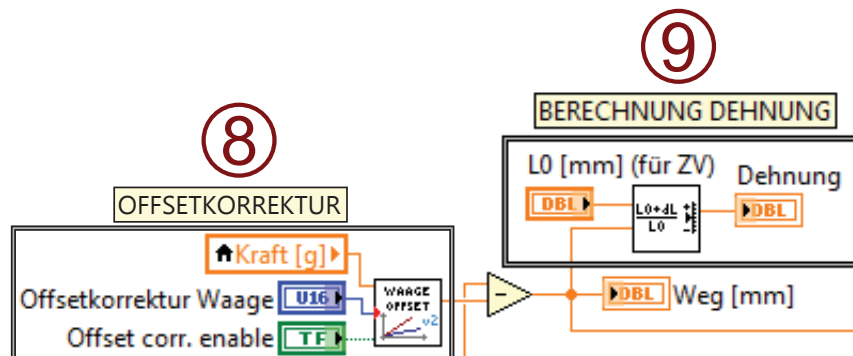


Abbildung E.8: LabVIEW® Blockschaltplan „Offsetkorrektur / Dehnung“

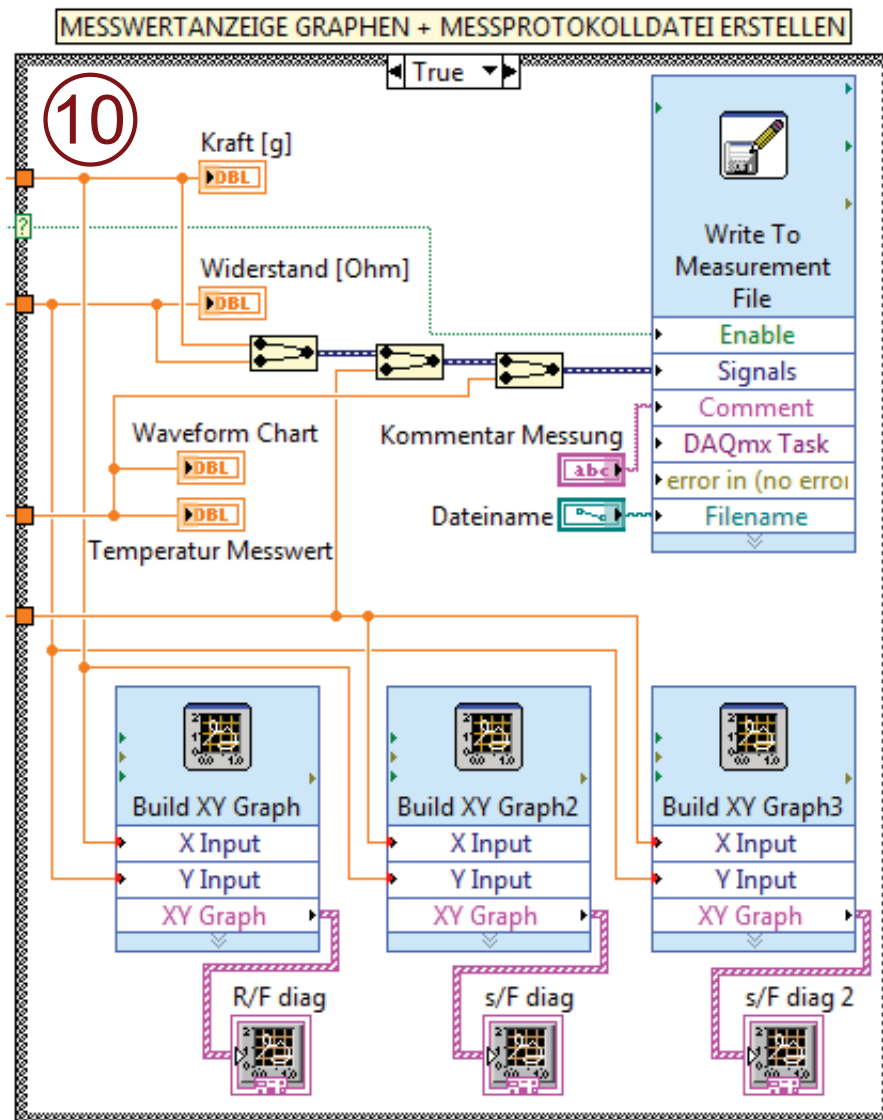


Abbildung E.9: LabVIEW® Blockschaltplan „Messwertanzeige Graphen, Messwertprotokoll erstellen“

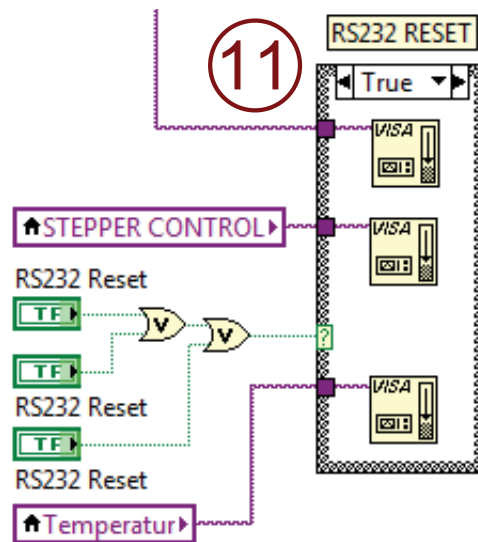


Abbildung E.10: LabVIEW® Blockschaltplan „Rücksetzen der seriellen Schnittstellen“



Abbildung E.11: LabVIEW® Blockschaltplan „Vorzeichen für Zug-/Druckversuch“

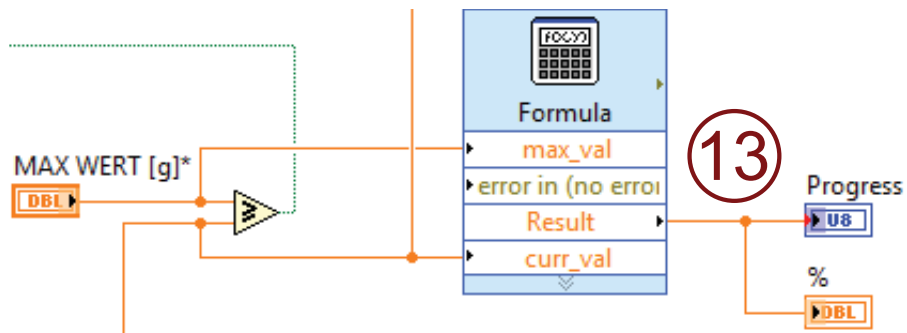


Abbildung E.12: LabVIEW® Blockschaltplan „Berechnung/Anzeige des Messfortschritts“



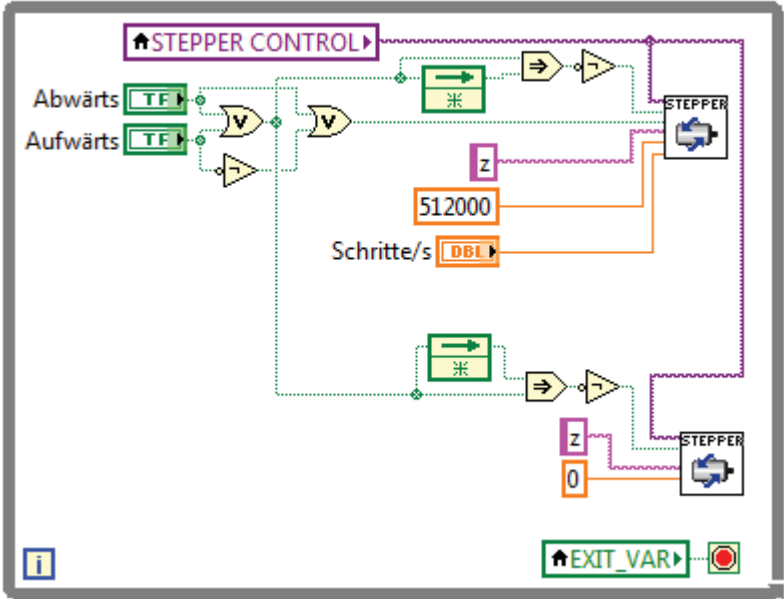


Abbildung E.13: LabVIEW® Blockschaltplan Schrittmotor manuelle Steuerung

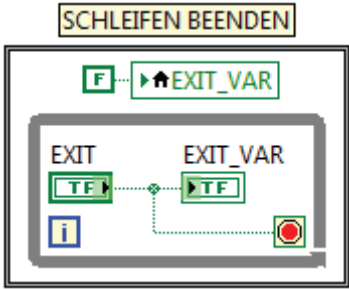


Abbildung E.14: LabVIEW® Blockschaltplan Schleifen beenden

## F LabVIEW<sup>®</sup> SubVIs Blockschaltpläne und Symbole

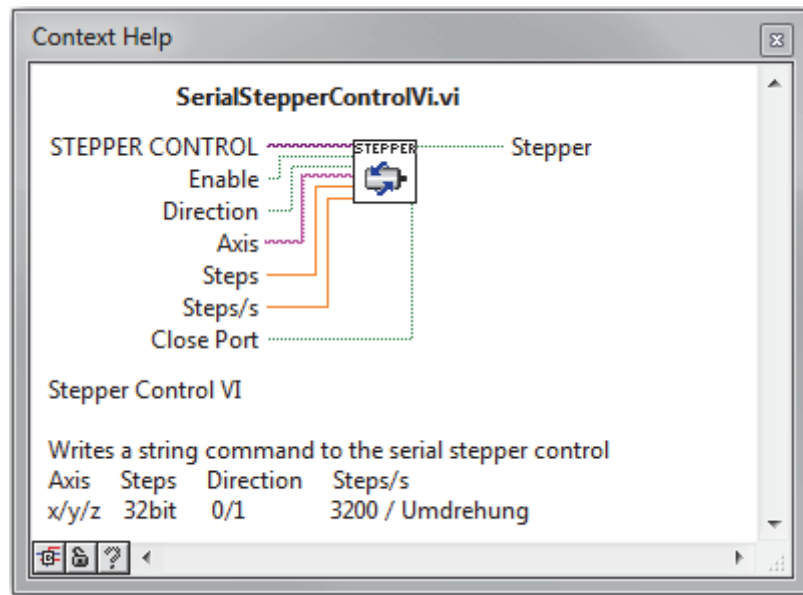


Abbildung F.1: LabVIEW<sup>®</sup> SubVI „Stepper Control“ Symbol

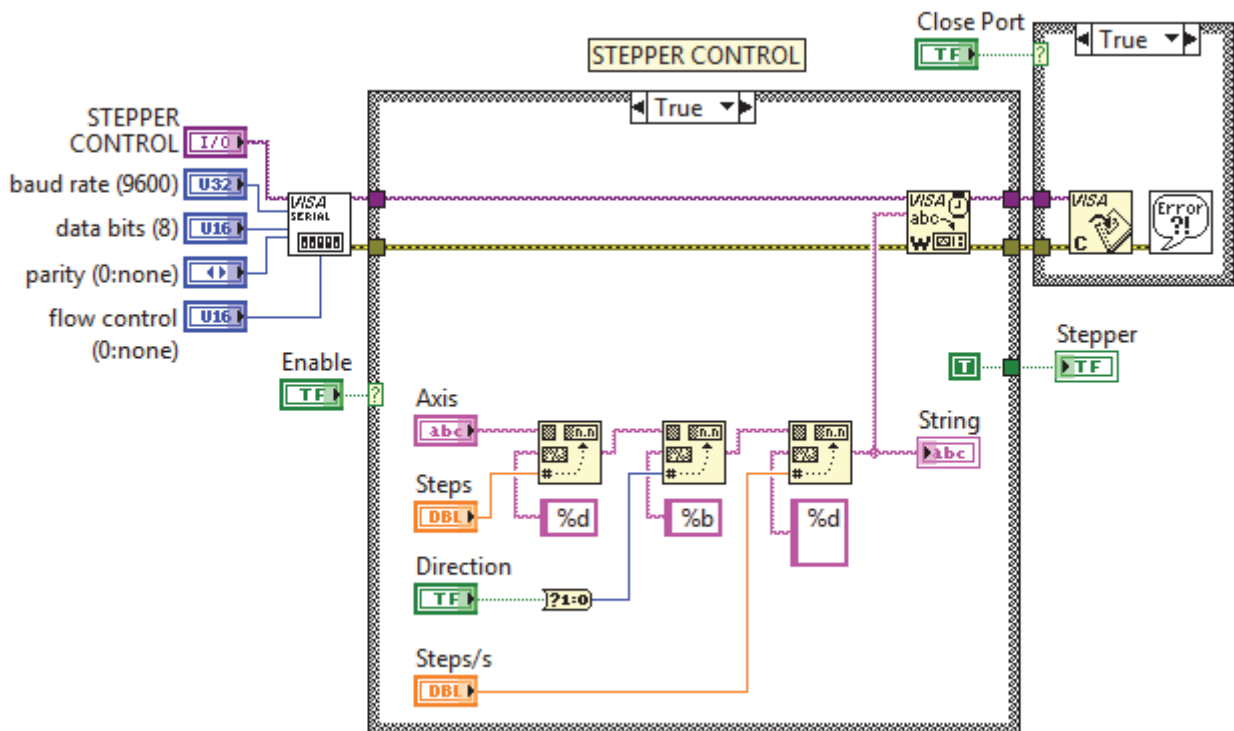


Abbildung F.2: LabVIEW<sup>®</sup> SubVI „Stepper Control“ Blockschaltplan

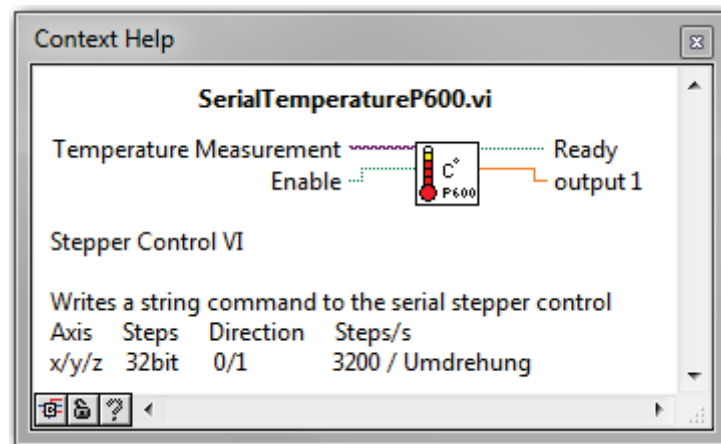


Abbildung F.3: LabVIEW® SubVI „SerialTemperatureP600“ Symbol

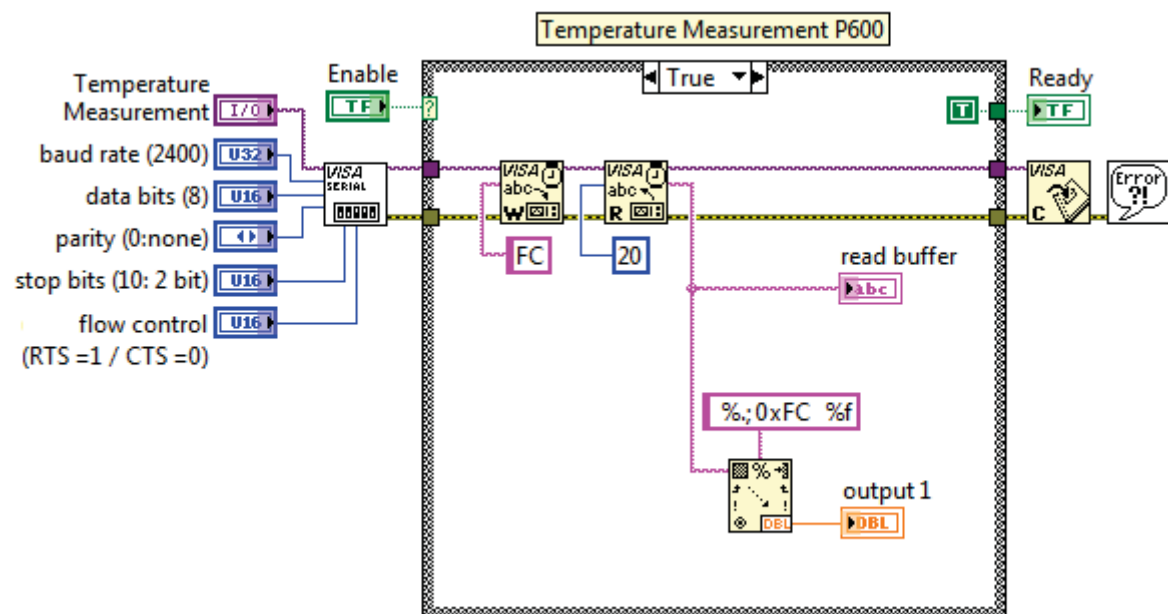


Abbildung F.4: LabVIEW® SubVI „SerialTemperatureP600“ Blockschaltplan

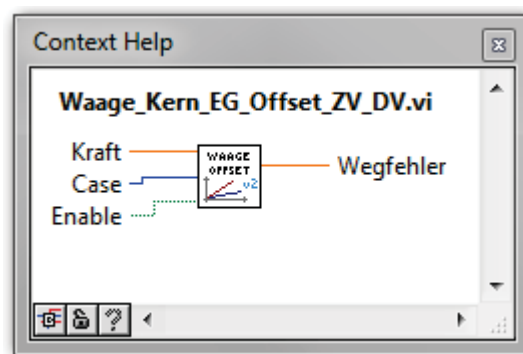


Abbildung F.5: LabVIEW<sup>®</sup> SubVI „Offsetkorrektur“ Symbol

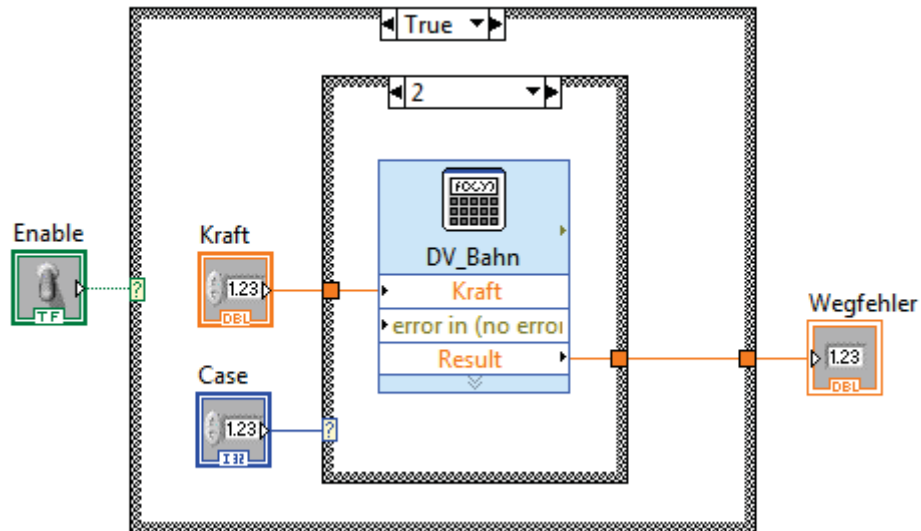


Abbildung F.6: LabVIEW® SubVI „Offsetkorrektur“ Blockschaltplan

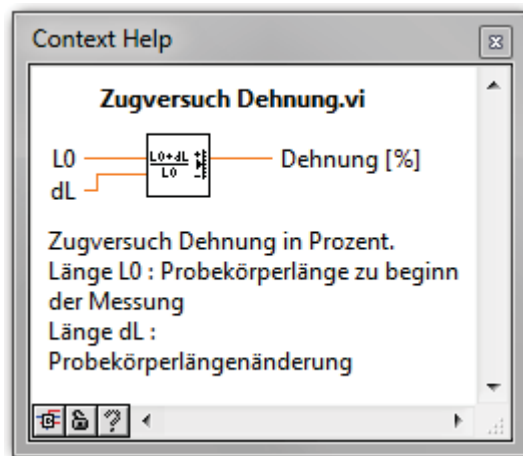


Abbildung F.7: LabVIEW® SubVI „Dehnung ZV“ Symbol

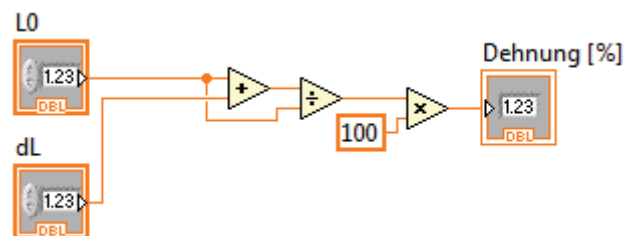


Abbildung F.8: LabVIEW® SubVI „Dehnung ZV“ Blockschaltplan

## G Kuka Leichtbauroboter-Struktur Abmessungen

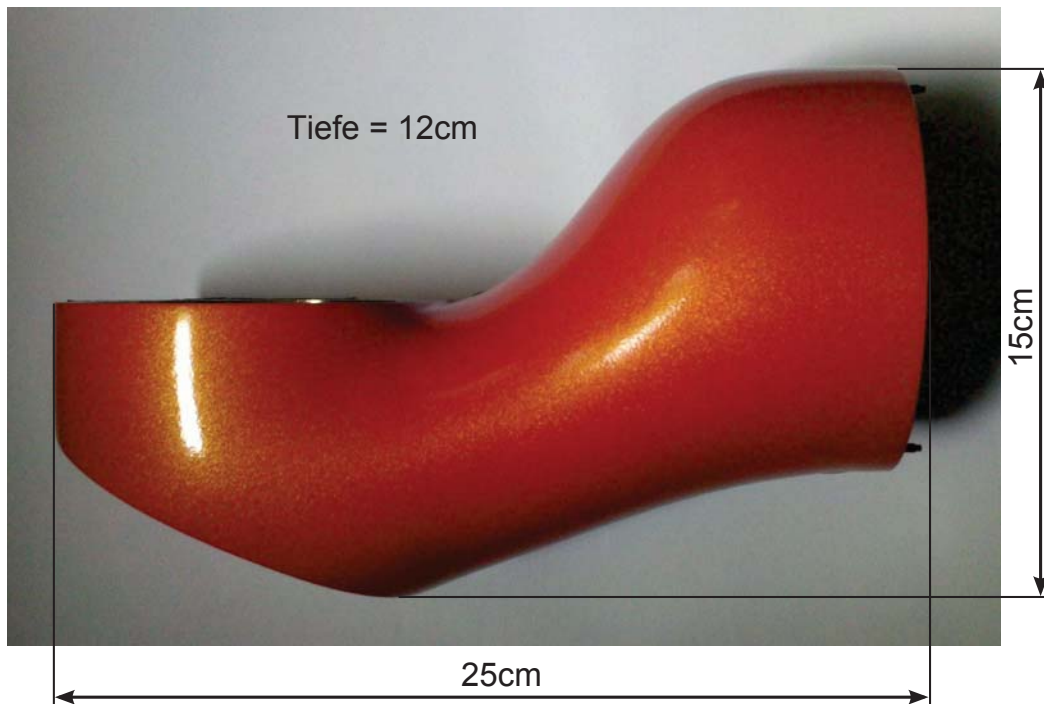


Abbildung G.1: Kuka LBR Struktur Abmessungen

## H Internationale Vergleichstabelle für Baustähle

Deutschland				Temp	USA AISI	Frankreich AFNOR	England BS	Italien UNI	Schweden SIS	Japan JIS
WkNr	DIN	alt	Typ							
1.0035	Fe 310-0	St 33	Baustahl		A 283 Gr A	A 34-2 NE	Fe 310-0	Fe 320	1300	-
1.0037	Fe 360 B	St 37-2	Baustahl		1015	E 24-2	Fe 360 B	Fe 360 B	1311	STKM 12 A
1.0044	Fe 430 B	St 44-2	Baustahl		1020	E 28-2	Fe 430 B	Fe 430 B	1412	SM 400 A
1.0050	Fe 490-2	St 50-2	Baustahl		A 570 Gr 50	A 50-2	Fe 490-2	Fe 490	1550	SS 490
1.0060	Fe 590-2	St 60-2	Baustahl		A 570 Gr 65	A 60-2	Fe 590-2FN	Fe 590	1650	SM 570
1.0070	Fe 690-2	St 70-2	Baustahl		-	A 70-2	Fe 690-2FN	Fe 690	1655	-
1.5622	14 Ni 6	-	Baustahl, kaltzäh	-100°C	A 350-LF 5	16 N 6	-	14 Ni KG	-	-
1.5662	X 8 Ni 9	-	Baustahl, kaltzäh	-200°C	A 350-LF 3	9 Ni 490	3603-509LT	X 10 Ni 9	-	SL 9 N53
1.5680	X 12 Ni 5	-	Baustahl, kaltzäh	-150°C	2515	Z 18 N5	-	-	-	-
1.5423	16 Mo 5	-	Baustahl, warmfest	+530°C	4520	-	-	16Mo5KG	-	SB 450 M

Abbildung H.1: Internationale Vergleichstabelle für Baustähle [12]

# Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Bachelorarbeit

## **„Konzeption und Aufbau eines Prüfstandes für taktile Sensoren“**

selbständig erstellt und keine anderen als die angegebenen Hilfsmittel und Quellen verwendet habe. Die aus fremden Quellen (einschließlich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht worden. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.

Oberpfaffenhofen, den 13.06.2012

---

Christian Leupolz

## Sperrvermerk

Die vorliegende Bachelorarbeit beinhaltet vertrauliche Informationen. Die Weitergabe des Inhaltes der Arbeit und eventuell beiliegender Zeichnungen und Daten im Gesamten oder in Teilen ist grundsätzlich untersagt. Es dürfen keinerlei Kopien oder Abschriften - auch nicht in digitaler Form - gefertigt werden.